THE UNIVERSITY
of EDINBURGH

This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e.g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

This work is protected by copyright and other intellectual property rights, which are retained by the thesis author, unless otherwise stated.
A copy can be downloaded for personal non-commercial research or study, without prior permission or charge.
This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author.
The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.
When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.

Efficient Techniques for Streaming Cross Document Coreference Resolution

Luke Shrimpton

Master of Philosophy

School of Informatics
University of Edinburgh

2017

Abstract

Large text streams are commonplace; news organisations are constantly producing stories and people are constantly writing social media posts. These streams should be analysed in real-time so useful information can be extracted and acted upon instantly. When natural disasters occur people want to be informed, when companies announce new products financial institutions want to know and when celebrities do things their legions of fans want to feel involved. In all these examples people care about getting information in real-time (low latency).

These streams are massively varied, people's interests are typically classified by the entities they are interested in. Organising a stream by the entity being referred to would help people extract the information useful to them. This is a difficult task: fans of 'Captain America' films will not want to be incorrectly told that 'Chris Evans' (the main actor) was appointed to host 'Top Gear' when it was a different 'Chris Evans'. People who use local idiosyncrasies such as referring to their home county ('Cornwall') as 'Kernow' (the Cornish for 'Cornwall' that has entered the local lexicon) should not be forced to change their language when finding out information about their home.

This thesis addresses a core problem for real-time entity-specific NLP: Streaming cross document coreference resolution (CDC), how to automatically identify all the entities mentioned in a stream in real-time.

This thesis address two significant problems for streaming CDC: There is no representative dataset and existing systems consume more resources over time. A new technique to create datasets is introduced and it was applied to social media (Twitter) to create a large (6M mentions) and challenging new CDC dataset that contains a much more variend range of entities than typical newswire streams. Existing systems are not able to keep up with large data streams. This problem is addressed with a streaming CDC system that stores a constant sized set of mentions. New techniques to maintain the sample are introduced significantly out-performing existing ones maintaining 95% of the performance of a non-streaming system while only using 20% of the memory.

Acknowledgements

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Luke Shrimpton*)

Table of Contents

Chapter 1

Introduction

Information is constantly being produced at an astounding rate: mainstream news websites each post an average of 300 stories a day (Meyer, 2016). On the social media platform Twitter, around 300M posts are made per day (Edwards, 2016). People need this information organised if they are to keep up to date with their interests. Organising information by entities is a convenient way to do this; people already use entities to find information: 71% of search queries contain a named entity (Guo et al., 2009). New information is very important, shown by the increase in search queries after news breaks in Figure 1.1. People go out of their way to seek out more information about an entity after news about it breaks.

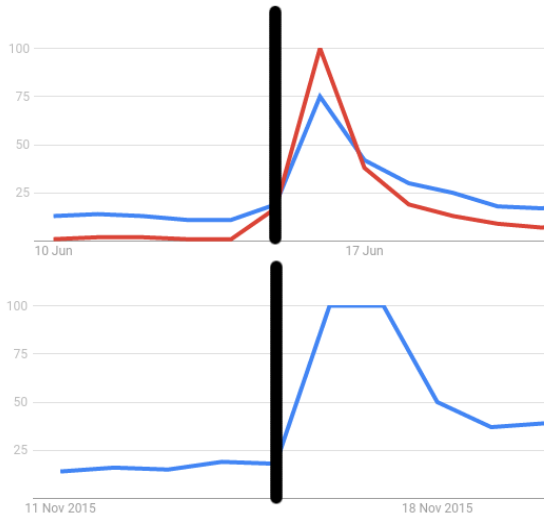

Figure 1.1: The relative search frequency from Google Trends. Top) Search terms 'Taylor Swift' (red) and 'Tom Hiddleston' (blue) the black line indicates when the news that they started dating broke. Bottom) Search term 'Ballast Point' (a small brewery) where the black line indicates when news about their buyout broke.

People turn to social media to keep up to date with entities that interest them (particularly celebrities); 20% of social media users cite it as a key reason for using social media (McGrath, 2015). Social media allows people to personalise the news they receive by following accounts specific to entities they are interested in. 88 of the 100 most followed accounts on Twitter are official entity-specific accounts (Counter, 2016) such as 'Katy Perry' and 'Real Madrid FC'. This demonstrates the popularity of entity-specific news. There are also many fan accounts that re-post existing content with new commentary. These accounts can have huge numbers of followers: For example, the One Direction Twitter fan account @STYLATORARMY has 1.3M followers, and the Kardashians Instagram fan account @kuwtkgirls has 1.2M followers.

People clearly want real-time entity specific information, although not all entities have sufficient interest for people to curate these entity-specific accounts. All these entity-specific accounts are curated by humans and require a significant amount of effort to maintain. Some people spend hours a day maintaining their fan account (Holmes, 2016).

Entity-level NLP (such as: information extraction or entity tracking) should not rely on the text used to reference the entity to uniquely identify the entity. This can cause significant problems. For example: Figure 1.2 shows searches about three British people who died in European plane crashes in 2015/2016. The name that produced the largest spike in search traffic was 'Richard Osman' which is also the name of a well known British TV presenter not involved in the crash.

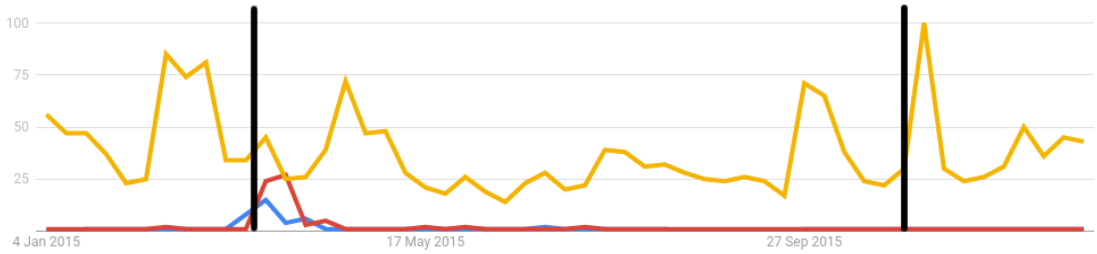

Figure 1.2: The Google Trends relative search frequency for the names 'Martyn Matthews' (blue), 'Paul Bramley' (red) and 'Richard Osman' (yellow). The first black line indicates the 2015 Germanwings crash. The second line indicates the EgyptAir 804 crash.

This spike is partly due to people thinking the celebrity named 'Richard Osman' had died. News reports typically quote press releases by providing a name and leaving the reader to identify the person referenced. It is reasonable for people to think the

news was referring to a well-known person, leading to people misunderstanding the story because they misunderstood the reference. Entity-specific NLP systems should not make those mistakes: Instead it should resolve the references before further reasoning about the entity (for example: populating a knowledge base).

It is important for all references to be resolved. There are far more less popular entities than the few popular entities mentioned in the mainstream news (the long tail). These entities are still important. Some people want the latest information about local politicians or businesses that are only important in small regions. Such information is far more likely to be found on social media. By only resolving the popular entities, people who want information about local entities will receive no benefit from entity-specific NLP technologies. They stand to gain the most as it can be more difficult to keep up to date with the less-popular entities that fewer people are writing about.

This thesis addresses a core problem for real-time entity-specific NLP: uniquely identifying every entity referenced in a continuous stream of documents. The remainder of this chapter formalises this problem, introducing cross document coreference resolution and demonstrating that social media provides a challenging domain with information about a broad range of entities.

1.1 Referencing a Entity

The task of identifying the entity for each reference is known as coreference resolution. The sequence of terms used to reference an entity (a **mention**) is extracted and the mentions that refer to the same entity are identified (**coreference resolution**).

Mentions are typically noun phrases that can be extracted using a shallow parse. There are three types of mentions with decreasing degrees of ambiguity:

1. **Pronoun:** Mentions such as ‘they’ and ‘it’ can refer to any entity, though gendered pronouns typically limit their reference to people of a specific gender.

2. **Common Noun:** Mentions such as ‘the doctor’ and ‘the bank’ refer to either classes of entities or a specific entity within that class.

3. **Proper Noun:** Mentions such as ‘John Smith’ and ‘Edinburgh’ refer to a unique entity. Resolving these mentions is still difficult as the text used to refer to the entity is often non-unique. For example: There are many people named ‘John Smith’ in the world and ‘Auld Reekie’ is another name for Edinburgh.

The least ambiguous case, proper nouns, are still very challenging. Names are not unique: the mention text 'John Smith' could refer to at least 197 different people according to Wikipedia. Without further contextual information, for example knowing his job ('John Smith is a botanist') it is impossible to determine if two mentions of 'John Smith' refer to the same entity. This is further hindered by alternative names as Table 1.1 demonstrates entities can have many alternative mention texts.

Mention Text	**Explanation**
Beyoncé	Canonical name of pop star
Queen Bey	Nickname that is often used by fans
Beyoncé Knowles-Carter	Full name
🐝	Fans use the bee emoji to refer to her
Edinburgh	Canonical name of British city
'Burgher	Colloquial nickname created by shortening original name
Auld Reekie	Historical nickname still in use today
Edynburghe	Historical spelling
Google	Canonical name of large tech company
Alphabet	Name of parent company, sometimes used mistakenly
GOOG	Stock code for Google
Big G	Nickname derived from IBM's nickname as Big Blue

Table 1.1: Various alternative names and their explanations

1.2 Cross Document Coreference (CDC)

Coreference resolution is typically broken down into two tasks: Within Document Coreference (**WDC**) and Cross Document Coreference (**CDC**).

WDC treats each document independently, focusing on resolving ambiguity caused by the use of common noun and pronoun mentions such as 'the doctor' and 'she'. WDC is a well-studied problem and has been the subject of the CoNLL shared task in 2011 and 2012 (Pradhan et al., 2011, 2012). Existing techniques are quite varied, though popular techniques involve using machine learning to build a mention similarity measure and then cluster (Soon et al., 2001; Ng and Cardie, 2002; Bengtson and Roth, 2008; Durrett and Klein, 2013) or use linguistically motivated rules for building

clusters (Raghunathan et al., 2010). The biggest challenge for modern WDC systems is resolving common noun references; these require knowledge about the entity being referred to. For example: identifying that 'the president' is referring to the same entity as 'Obama' requires the knowledge that 'Obama' fulfilled the role of 'president'. Proper noun references are typically not a problem for WDC systems as it is very rare for a document to contain two identical proper noun references to different entities or two proper noun mentions of the same entity with very different mention texts.

CDC resolves mentions from multiple documents, typically focusing on proper noun mentions. These references can be quite ambiguous. There are two types of ambiguity CDC must resolve:

1. **Synonymy:** There can be multiple mention texts for a single entity. Spelling mistakes and mention shortening/lengthening (for example: adding/removing a title/first name) are common although alternative names can also exist for many reasons such as a person getting married, a corporate re-branding and locations changing names.

2. **Polysemy:** The same mention text can be used for multiple entities. This is particularly common for person entities; if a common first name is used to mention an entity, it is very ambiguous which entity is being mentioned.

To determine if two proper noun mentions refer to the same entity, two types of information are used:

1. **Mention Text Similarity:** A measure of mention text similarity (for example: edit distance) is a useful feature for detecting spelling errors or if an orthographically similar mention text is used.

2. **Contextual Similarity:** If two mentions use the same mention text but refer to different entities, mention text similarity is not sufficient. A measure of the contextual similarity is required, for example: Identifying that they are referred to in very different topics (sports vs politics).

While WDC is not considered a 'solved' NLP task state of the art approaches are well developed and efficient. As each document is independent, the amount of mentions per WDC problem is bounded by the length of the document. There are less scalability problems as WDC can be parallelised trivially. CDC has no natural dataset size limit: a CDC system could have to process any amount of mentions. Ensuring

any amount of mentions can be processed in real time is a non-trivial problem that is addressed in this thesis.

1.3 Cross Document Coreference for Social Media

The mainstream media only report on a narrow range of entities, the popular ones. Social media contains a much broader range of information. While often belittled for being full of teenagers talking about 'Justin Bieber', popular social media platforms have very broad coverage. Their users include teenagers interested in celebrity gossip, international communities of NLP researchers and local special-interest communities such as 'foodies'. The range of events extracted from Twitter demonstrates its diversity (Ritter et al., 2012). Mainstream events such as the Democratic National Convention are identified amongst less popular events such as the release of the new Nvidia graphics card and the launch of a new Harry Potter book.

The very informal nature of the text leads to lots of idiosyncrasies and spelling mistakes, increasing the ambiguity of mentions. Figure 1.3 demonstrates some of the novel ways people mention entities in social media.

#1Dreamboy2today i just beat up taylor swift
YEY FOR TAY TAY!
My mom says I'm a better singer than Tswizzle lol

Andy - what do you want for your birthday? The Italian Open title please, beating Djokovic. No problem.
If Murray will ever win the French open, this has to be the year. Easiest route to the final he's ever had.
Australian Open: Djokovic Beats Murry, Wins 6th Australian Open Title

We are SUPER excited to have @martinamcbride LIVE on our Fbook page! Tune in at 2:15pm
$FB is now the most important stock for hedge funds - GS
facebook has broken successful (as far as profit, at least) websites based on how they tweak their algorithm, so yes, it's an issue

Figure 1.3: Demonstrating alternative mentions to various entities with the mention text in red. Some of the 'Taylor Swift' and 'Facebook' mentions use some very novel mention texts, 'Andy Murry' is mentioned using only his first or second name which is very ambiguous.

Some communities have their own unique CDC problems: fandom communities (groups of people who are very interested in particular celebrities or media franchises) often come up with novel names for people or pairs of people (often to express their approval of a particular romantic relationship). Figure 1.3 shows some of the novel names for 'Taylor Swift' though the pair 'Taylor Swift' and 'Harry Styles' were often referred to as 'Haylor' and the more recent pairing of 'Taylor Swift' and 'Tom Hiddleston' is referred to as 'Hiddleswift'. These mentions are referencing two people at once, which breaks the normal assumption that each term can only be used to reference a single entity.

For comic book communities, the names of superheroes and their secret identities provide challenging CDC problems. For example, it is clear that 'Clark Kent' and 'Superman' refer to the same person although they use completely different mention texts and potentially very different contexts as shown in Figure 1.4. With some comic books, there is the added problem of re-boots and alternative realities. For example: the comic-book based TV show 'The Flash' where there are multiple people named 'Barry Allan' (the secret identity for 'The Flash') from different realities; should these

people be considered the same entity?

```
Clark Kent. Great at alien things. Not so great with human emotions and stuff.
Poor Chloe
How does Superman's hair stay nice AF even after being hit and plummeted through
buildings?
```

Figure 1.4: Demonstrating mentions of 'Clark Kent' and 'Superman' where the mentions are in very different contexts.

Sports also introduce difficult CDC problems: when an entity is playing a game, the ambiguity of the reference can increase if the author assumes the entity is salient to the rest of their followers. The author believes the temporal context that the game is currently happening is sufficient for someone reading the mention to resolve it. Figure 1.5 shows mentions of the tennis player Roger Federer made while he was playing.

```
Go Federer Go!!!
come on Federer!
7-6 to Roger
```

Figure 1.5: Demonstrating very ambiguous mentions of 'Roger Federer' while he was playing a game.

In some cases, the CDC problem can be easier, for example: The Screen Actors Guild (Guild, 2016) has a policy that no two members can have the same name, so popular actors such as 'David Tennant' and 'Emma Stone' had to change their names from 'David McDonald' and 'Emily Stone' respectively.

1.4 Summary

To deal with the ever increasing amount of information produced, people turn to entity-specific information either through searching or following entity-specific social media accounts. For NLP systems to organise information by entity, they need to resolve every reference to an entity (perform CDC)

To provide people with information on a broad range of entities, social media streams should be analysed. CDC on social media is a challenging task due to the

idiosyncrasies and varied language used, resulting in many novel mention texts compared to the mainstream news. The massive variety of domains discussed on social media also introduces novel challenges although it provides a much broader range of entity-specific information than mainstream news.

The following problems are addressed in this thesis:

1. **Dataset Construction and Evaluation:** Existing techniques to create datasets either introduce a significant amount of bias or require a large amount of manual effort, limiting the size of datasets it is possible to create. This thesis introduces a new technique for efficiently annotating a large dataset and demonstrates the best evaluation technique to use. This dataset construction technique is applied to a large corpus of tweets to create a new large social media dataset.

2. **The Streaming Model:** Existing systems will always reach a point where they use more resources (time and memory) than available. This thesis presents the first CDC system that works with constant resources (constant amount of memory and constant amount of latency). The techniques presented address a number of problems specific to CDC, resulting in a significant performance improvement over existing techniques. Maintaining 95% of the performance of a non-streaming system while only using 20% of the memory.

Chapter 2

Background

Previous CDC research can be broken down into three main topics: Modelling mention text similarity, sources of contextual information and how to scale CDC.

2.1 Mention Text Similarity

Mention text similarity will help identify misspellings, mention shortening/lengthening as demonstrated in table 2.1.

Mention Text	**Description**
Roger Federer	Full name
Mr Roger Federer	Full name with title (Mention Lengthening)
Federer	Second Name (Mention Shortening)
Federer, Roger	Terms Reversed
Rog Fed	Misspelling
Fedex	Common nickname (Spelling Error)

Table 2.1: A range of mention texts used to refer to the entity 'Roger Federer.

While there are many approaches to mention text similarity they can be broken down into three catagories (Cohen et al., 2003):

- **Edit distance like similarity**[1]**:** Measures similarity between two strings by comparing characters and their ordering. Examples of these include Levenshtein distance and Jaro-Winkler similarity (similar to edit distance though weights the

[1]Similarity measures that use characters and their positions but are not Levenshtein edit distance

first characters higher). These techniques fail when mention texts are re-ordered or there is mention shortening/lengthening, although they are good at resolving spelling errors. When combined with phonetic encoding algorithms such as metaphone (Philips, 2000) terms that sound similar are matched.

- **Term-based similarity:** Measures similarity between (sometimes weighted) sets of terms. Examples of these include tf-idf weighted cosine similarity and jaccard similarity. These techniques fail when there are spelling errors (terms are no longer identical) though work well when terms are re-ordered or there is mention shortening/lengthening.

- **Hybrid Similarity:** As the previous two approaches have complementary strengths and weaknesses, hybrid techniques such as Soft-tf-idf (Cohen et al., 2003) combine both approaches, so mention texts with both spelling errors and shortening/lengthening can be resolved.

Overall, term-based similarity outperformed edit distance like similarity demonstrating the importance of ignoring term ordering and accounting for mention shortening/lengthening, although hybrid techniques work the best.

The hybrid techniques soft-tf-idf (better thought of as soft-cosine similarity) uses a term similarity measure to allow soft matches between terms when computing cosine similarity. Standard edit distance and term-based similarity measures typically require no parameters. Hybrid techniques are at least parameterised by the term similarity measure used, which can have a large effect on both efficiency and effectiveness.

More complex techniques can be deployed if training data is available: Supervised machine learning can be used to create a mention text similarity measure: either learning to combine existing similarity measures (Cohen and Richman, 2001) or learning an edit distance style measure (Ristad and Yianilos, 1998).

Mention texts can evolve over time, due to copy errors or pressure to come up with a mention that sounds nice/edgy. For example: ‘Taylor Swift’ is sometimes called ‘T-Swift’ which has led to the nickname ‘T-Swizzle’. Edit distance may be able to model the differences that occur over one generation as they are often minor, though a generative model can be learnt to model the entire process(Andrews et al., 2014) allowing mentions of ‘T-Swizzle’ to be correctly resolved with mentions of ‘Taylor Swift’.

Techniques such as Levenshtein distance are very inefficient, they require solv-

ing an optomisation problem for each pair of strings (the minimum amount of insertions/deletions required to transform the strings).

Character n-gram similarity can be used to efficiently/effectively measure mention text similarity (Zobel and Dart, 1995; Nguyen et al., 2014; Moreau et al., 2008). This involves representing a mention text as a set of n-grams (as shown in Figure 2.1) then using a set-based similarity measurement (such as Jaccard, DICE or cosine similarity). This is effective because in using n-grams, the characters and some of the character ordering information are maintained. If there are some character deletions/insertions, only some of the n-grams in the set will be different. Terms are not maintained so term-ordering or mention shortening/lengthening will not severely impact them much like a hybrid technique. This technique is used later in the thesis.

```
'Roger Federer' => (rog,1), (oge,1), (ger,1), (er ,1), (r f,1), ( fe,1),
(fed,1), (ede,1), (der,1), (ere,1), (rer,1)

'Fedex' => (fed, 1), (ede, 1), (dex, 1)
```

Figure 2.1: Demonstrating how terms are deconstructed to 3-gram indicator vectors.

2.2 Contextual Similarity

Contextual similarity is any similarity based on information about the mentions other than the mention text and can take various forms. It is essential to determine if two identical mention texts refer to different entities.

Document similarity is the most common measure of contextual similarity (Bagga and Baldwin, 1998b; Gooi and Allan, 2004; Niu et al., 2004; Mann and Yarowsky, 2003; Chen and Martin, 2007; Baron and Freedman, 2008; Dutta and Weikum, 2015b,a; Singh et al., 2011; Rao et al., 2010). Documents about similar topics are likely to mention similar entities. Figure 2.2 shows two tweets that mention someone named 'Roger'; the first tweet is clearly about a tennis player while the second is about a TV character, clearly indicating they are referring to different entities. As this technique is both effective and efficeint (requires no pre-processing) it was used in this theis.

```
this tennis is brilliant!  come on roger!  #USOpen

watching American Dad, I only watch this show for Roger.
```

Figure 2.2: Demonstrating two tweets that mention an entity using the same mention text though the tweets are about different topics, mention text shown in red.

Using the entire document includes information about other entities which may not be relevant to the mention being resolved (the target mention). Using only a 'snippet' of terms near the target mention works better than using the entire document (Bagga and Baldwin, 1998b), although using terms from the sentences that contain a mention of the entity (identified using WDC) has also been shown to work well.

The next most common source of contextual information is the other mention texts in the document (Chen and Martin, 2007; Mann and Yarowsky, 2003; Niu et al., 2004; Baron and Freedman, 2008; Haghighi and Klein, 2007; Dutta and Weikum, 2015b,a). While a mention text is ambiguous by itself, often the only information required to correctly resolve a mention is one other mention. For example: Figure 2.3 shows two tweets about drummers named 'Roger Taylor'; at a casual read, they appear to be about the same person although the tweets contain a mention of the band name, which indicates they are referring to two different entities.

```
Just staring at Roger Taylor videos play the drums.  Hawt with his long blonde
hair.  #KillerQueen #Queen
@duranduran my favorite drum by roger taylor is on the wild boys song in the
beginning.  It's so cool.
```

Figure 2.3: Demonstrating two tweets that mention someone named 'Roger Taylor' mention texts are shown in red and important context terms are shown in blue.

Using other mention texts in the document suffers from the same problem as document similarity. It is difficult to determine if another mention text is relevant to the target mention. Again, it is common to use 'snippets' or sentences to identify relevant mention texts although information extraction techniques can be used to identify mention texts that are in a relationship with the target mention (Mann and Yarowsky, 2003; Baron and Freedman, 2008).

Event coreference: Identifying verbs that refer to the same event can help resolve entity coreference (and vice versa) (Lee et al., 2012). For example in Figure 2.4, if the term 'won' is referring to the same event, then the mentions of 'Roger' and 'Federer' are referring to the same entity.

```
Hmmm.. So Roger won his Q/F match ... Did I enjoy it? Heck NO ! Awful, Awful
tennis!
Ok! I missed Qtr Final: Roger Federer vs Jo-Wilfried Tsonga. Federer won!
Heard it was a good match. Sad sad...
```

Figure 2.4: Demonstrating tweets where identifying coreferent verbs (shown in blue) can help resolve the mention texts shown in red.

2.3 Other CDC Systems

The only work on cross lingual CDC (Green et al., 2012) demonstrated how to transliterate/translate mention texts and context with both high-resource languages (machine translation systems are available) or low resource languages (only documents of similar topics are required).

CDC systems have no prior knowledge whereas entity linking has some prior knowledge, often in the form of Wikipedia pages (or a structured knowledge base such as Freebase). Instead of clustering the mentions, they are linked to an entry in the knowledge base, either reporting NULL or creating an entry if it does not exist.

The knowledge base provides some reliable knowledge about each entity unlike most documents where it is difficult to determine information about the target mention; most of the content on a Wikipedia page is directly relevant to the entity. Links between entities in the knowledge base can also be used to provide important information (Cucerzan, 2007).

Previous research demonstrated how using a knowledge base can help with CDC (Dutta and Weikum, 2015a). Entity linking and CDC can be performed jointly, each task helping to correct the other's mistakes. As knowledge bases only contain information about popular entities they can hinder performance for less popular (long tail) entities (Dutta and Weikum, 2015b), the likes of which are very common on social media.

2.4 Scaling CDC

Efficient CDC resolution requires efficient models, and most CDC systems use a basic hierarchical agglomerative clustering model that involves $O(n^2)$ mention pair similarity computations (Bagga and Baldwin, 1998b; Mann and Yarowsky, 2003; Gooi and

Allan, 2004; Chen and Martin, 2007). Some approaches have reduced complexity by pre-clustering such as: Only clustering mentions with similar mention texts (Baron and Freedman, 2008) or similar contexts (Lee et al., 2012). Alternatively, approximate clustering algorithms can be used (Dutta and Weikum, 2015b). One effective technique to make CDC resolution scale is to reason with sets of mentions that are similar. Figure 2.5 shows how each entity can be represented by a set of sub-entities which contain mentions.

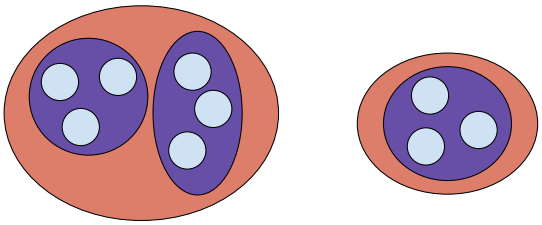

Figure 2.5: Demonstrating two entities (red) with sub entities in purple and mentions in blue.

CDC with this model involves moving mentions between sub-entities and sub-entities between entities until a good solution is found (Singh et al., 2011). A similar approach involves representing entities/sub-entities/mentions as roots/nodes/liefs in trees (Wick et al., 2012).

2.4.1 Streaming CDC

These existing approaches are designed to run on large *static* datasets; they are not appropriate for processing streams in real-time. As the stream progresses, the cost (in space and time) of processing a new item increases until it becomes infeasible to process new items: Either the hardware runs out of memory or the latency becomes too high. A system should be able to process an infinite stream with constant/bounded latency.

To ensure a system is able to process an infinite stream in real-time, it must conform to the streaming model of computation (Muthukrishnan, 2005). This sets out two constraints that must be satisfied:

- Each item is processed in a bounded amount of time: The system will never reach a point where it takes too much time (latency exceeds acceptable bounds) to process an item.

- Use at most a constant amount of memory: This ensures the amount of memory used will not grow to the point where all the memory is used and the system crashes.

Building systems that conform to this model can be very challenging; previously trivial tasks require novel solutions. For example, with frequency counting: Given an infinite stream of numbers between 0 and n, how should their frequencies be counted when only enough memory to store $n' << n$ counters is available? A standard solution to this problem (sketching) involves hashing each number to one of the n' counters and then incrementing/querying that counter. The frequencies are only approximations as hash collisions occur, although it is possible to compute a bound on the expected error[2].

The frequency counting problem demonstrates a core principle when developing systems for the streaming model of computation: Exact solutions are (almost always) not possible, so streaming systems must make approximations. By trading off performance (making approximations), systems are able to process significantly more data, faster.

More complex streaming systems are required to make more approximations. For example, with a sequential single link clustering system, each new item is compared to all previously seen items to identify its nearest neighbour. A streaming system (Petrović et al., 2010) can only store a limited amount of previously seen items which will lead to an approximation. To further decrease the latency, an approximate nearest neighbour technique can be used.

The only system for resolving mentions from a stream of documents does not conform to the streaming model of computation as memory usage increases with time (Rao et al., 2010). Their system performs an initial candidate retrieval using character skip bigrams to find entities with similar mention texts and then re-ranks the candidates using a measure of contextual similarity (tf-idf weighted document similarity). To form entities, a nearest neighbour based clustering technique is used, and the new mention updates the existing entity modifying its document representation and adding any new mention texts to its index. Their simple technique achieves logarithmic time/space complexity.

Most general purpose 'streaming' clustering systems that could be used to resolve CDC are not designed for the streaming model of computation. These techniques typically process items sequentially, though break at least one of the constraints of

[2]For an overview of common streaming algorithms see: `http://www.cs.dartmouth.edu/~ac/Teach/CS49-Fall11/Notes/lecnotes.pdf`

the streaming model of computation. For example, some systems only summarise the stream for clustering later (Chen and Tu, 2007) (requiring all the data to be store breaking the memory constraint). Some systems need time to reduce the memory used when memory runs low (Aggarwal et al., 2003; Charikar et al., 1997) introducing extra latency for some items. Other techniques involve converting a stream into batches of updates so the latency is no longer constant (O'callaghan et al., 2002). Systems that assume the memory is not constrained or where there is a more relaxed constraint such as allowing it to grow logarithmically (Aggarwal et al., 2003; Charikar et al., 1997) clearly do not conform to the streaming model of computation.

2.5 Summary

The types of information required to resolve CDC have been well studied. The main problem with existing CDC systems is efficiency. They do not scale to the infinite streams required to provide the information people want.

Streaming systems need to make approximations, but it is not clear what trade-offs need to be made to ensure the correct approximations are made when developing a streaming CDC system. The remainder of this thesis will tackle this and related problems.

Chapter 3

Dataset and Evaluation

Streaming CDC lacks a good quality dataset. The single existing dataset (Rao et al., 2010) contains only automatically introduced polysemy ambiguity, which is not representative of real language. Figure 3.1 outlines the ideal properties of a CDC dataset.

1. Both polysemy and synonymy ambiguity.
2. Natural (human-generated) mention texts.
3. No bias towards particular entities/mentions. For example: A dataset should not ignore entities for social reasons (not considered interesting by the mainstream news) or mentions for linguistic reasons (it was misspelled).

Figure 3.1: The three ideal properties of a CDC dataset

This chapter reviews existing techniques for creating a CDC dataset, showing that creating a representative dataset requires a lot of manual annotation.

The manual annotation of large CDC datasets is infeasible if all mentions are annotated (the current standard). If only a sample of entities are annotated, it becomes feasible. A technique for creating a large dataset by annotating a sample of entities is introduced. This technique is applied to a large collection of 52 million tweets to produce a new streaming CDC dataset.

When a sample of entities are annotated, existing evaluation measures are not directly applicable; rather they assume all (or all non-singleton) entities are annotated. A simple modification is proposed so datasets with partial annotations can be evaluated. This modification is experimentally demonstrated to be robust when evaluating a system using a dataset that contains partial annotations.

3.1 Dataset Construction

Existing techniques to construct a CDC dataset fail to produce a representative dataset; they break at least one of the conditions outlined in Figure 3.1. Most techniques follow the standard practice of annotating person named entity mentions and ignore non-annotated mentions.

3.1.1 Ambiguous Mentions

The first CDC dataset (Bagga and Baldwin, 1998b) chose a common full name: 'John Smith' and manually annotated every mention of it in a small document collection. All the mention texts are natural and there is only polysemy ambiguity. This techniques annotates a very limited set of mentions (must be a full name reference) and entities (must be an entity with a common full name). A similar dataset was created by searching for ambiguous names on Google and annotating the first 100 document for each search result (Chen and Martin, 2007).

3.1.2 Person-X

The person-x methodology is the most popular method for constructing a CDC dataset (Gooi and Allan, 2004; Pedersen et al., 2005; Mann and Yarowsky, 2003; Singh et al., 2011; Mann and Yarowsky, 2003; Rao et al., 2010). It involves automatically inserting polysemy ambiguity by rewriting an unambiguous mention as an ambiguous one: For examplem the mentions 'Taylor Swift' and 'Beyoncé' are each very likely to refer to a unique entity; hence, rewriting them both as 'Person-1' introduces polysemy ambiguity. This technique produces unnatural mention texts with only polysemy ambiguity and only annotates a subset of entities (can be mentioned using an unambiguous name) and mentions (unambiguous ones).

3.1.3 Wiki-Links

Hyperlinks to Wikipedia pages can be used to identify and annotate mentions in web documents. Text spans that are also hyperlinks to a Wikipedia page are likely to be a mention of the entity referred to by the Wikipedia page (Singh et al., 2012). The mention texts are not entirely natural as there is no guarantee that people put hyperlinks around a linguistic mention. Both polysemy and synonymy ambiguity are included.

There is significant entity/mention bias as only entities that have a Wikipedia page are annotated and people self select which mentions to annotate.

3.1.4 Manual Annotation

There are two datasets that were constructed using exhaustive manual annotation. A portion of the ACE-08 dataset (Strassel et al., 2008) (400 documents) was both WDC and CDC annotated though they only annotated 50 entities that were mentioned between 4 and 100 times for CDC. The EventCorefBank (Lee et al., 2012) dataset consists of 482 documents grouped into 43 topics where each topic was exhaustively annotated for both WDC and CDC effectively creating 43 small CDC datasets.

The different annotation techniques are summarised in table 3.1. The only existing streaming CDC dataset was annotated using the person-X methodology (Rao et al., 2010). It is clear that manual annotation is currently the only viable technique despite the time required to produce the datasets.

Annotation Technique	**Ambiguity**	**Mention Bias**	**Entity Bias**	**Streaming**
Ambiguous Name	S	+	+	No
Person-X	S	+	+	Yes
Wiki-Links	P/S	+	+	No
Manual	P/S	-	-	No

Table 3.1: Annotation techniques and their properties, P indicates the dataset includes polysomy and S indicates synonymy ambiguity. + indicates this bias exists, - indicates this bias does not necessarily exist.

3.1.5 Annotation Process

Streaming CDC research needs a representative dataset. A large number of documents need to be annotated for CDC, and efficiently in a way that ensures a minimal amount of compromises.

While this thesis focuses on a stream of tweets the annotation process described does not require the documents to be any particular type or organised into a stream.

The proposed annotation technique involves manually annotating *all* the mentions for a sample of entities (providing partial annotations). Unlike previous datasets, unannotated mentions will be kept. Removing unannotated mentions throws away any

mentions that may make correctly resolving the annotated mentions more challenging. Keeping unannotated mentions reduces the potential bias in the annotation technique as, if a mention/entity is not annotated, it can still influence system performance.

As mentions will be manually annotated, the mention text will be natural. The following two conditions must be met:

1. Mentions that are difficult to resolve due to synonymy and polysomy should be included.

2. Entities should be chosen in an unbiased way.

If entities are sampled uniformly at random (satisfying condition two), there will be no guarantee that polysemy exists as it is unlikely two random entities are mentioned using exactly the same mention text. If every mention of each entity is annotated, there is likely to be synonymy ambiguity if enough entities are annoated. By biasing the entities sampled, it is possible to ensure polysemy exists.

Only person entities are considered as sampling a diverse range of entities, ensuring polysemy ambiguity is easy due to the way people are typically mentioned using one of their names. We used the following procedure to sample entities:

1. Choose a common first name.

2. Annotate every mention of that name.

3. Discard entities mentioned only once.

If a common first name is chosen, the exact choice of name is unimportant: there will be a diverse set of entities mentioned using any common first name. Synonymy ambiguity should be present as all mentions (not just mentions using the common first name) are annotated. The limitations and bias introduced by this entity sampling technique and the person restriction is discussed in Section 3.1.6.

Once the entities have been chosen, the annotator then identifies all mentions of the entities in the dataset. This can be very difficult, particularly if the annotator is unfamiliar with the entity. To ensure all the mentions of the entity are correctly identified, the annotator observed the following procedure that was derived during initial annotation experiments:

1. Research the entity being referred to: If the entity has a Wikipedia page, read that and search the Internet for information about the entity.

2. Search the dataset for any alternative names that were identified during the initial research.

3. Inspect documents that contain terms with either a low edit distance or jaro-winkler distance from any of the terms used to mention the entity (identify any spelling errors).

4. Identify contexts in which the entity may be mentioned (either through previously annotated tweets or the initial entity research) and investigate tweets from those contexts. Contexts can include periods when the entity was mentioned frequently.

3.1.6 Limits of the Annotation Technique

By restricting a dataset to person entities, there is no restriction on the type of ambiguity (polysemy and synonymy) though there are important differences between how the various types of entities are mentioned. Differences between person, organisation and location entities and how they affect CDC are considered:

- **Canonical names:** With people, their canonical name is their full name in western culture it is typically 'first name surname name'. Organisations and locations have legal names. There are no rules governing the structure of a location and organisation name. This freedom may reduce the amount of polysemy for location and organisation entities.

- **Name Generation Influences:** With person entities, there is social pressure for a parent to choose either a unique or a currently popular first name. With organisations there is both economic and legal pressure for an organisation to have a unique name. When naming a place there is clearly pressure to re-use existing location names, this is shown by the number of locations in America named after somewhere in Europe. For organisation CDC polysemy is clearly not as much of a problem as person or location CDC.

- **Generation of Alternative Names:** Person names are often shortened (use their first/second name) or lengthened (add the title or middle name). Nicknames are common particularly for 'pop culture' entities. Locations and organisations are rarely lengthened due to their well-defined canonical name, though people often use nicknames for convenience. With organisations, initials and stock codes are

also often used. Typically, when a nickname is generated, it results in a name orthographically related to the original name that is 'nice' to say. Alternative person mentions are typically formed by dropping/adding terms whereas organisations and locations typically involve generating new terms. Some techniques (term-based similarity measures) may perform better on person CDC than location and organisation CDC.

- **Contextual Differences:** It is difficult to make any generalisations about the contexts in which different types of entities are mentioned. For all types of entities, there exist entities that are mentioned in many different contexts (for example: 'Obama', 'Facebook', 'London') and entities mentioned in very few contexts; typically, the more popular an entity is the more contexts it is mentioned in.

Ideally, a large uniform sample of entities would be selected; this ensures entities from a range of backgrounds (celebrities, business people, etc.), various popularities and unique, ambiguous or many alternative names.

In sampling using a common first name, the sample contains entities from a range of backgrounds and popularities; the only difference is there are no people with particularly unique names. Entities with unique names may introduce some interesting polysomy ambiguity (people may struggle to spell the names), but their mentions are likely to be fairly easy to resolve compared to entities with popular names. Sampling using a common name will produce a dataset that is harder than a uniform sample of person entities as mention text similarity is less effective.

The entity sampling technique involves discarding entities mentioned by the chosen name once. This introduces some popularity bias. Discarding these entities was done to reduce the amount of entities for the next annotation phase. Entities mentioned only once using the chosen first name are not likely to be mentioned frequently by other names. Determining if a mention is exophoric (does not refer to another entity in the dataset) is difficult, particularly with a large dataset; hence, filtering out entities that are exophoric early significantly reduces the time spent annotating.

Overall, the entity sampling technique introduces a relatively small and acceptable amount of bias given how efficiently a large dataset can be annotated.

3.1.7 Tweet CDC Dataset

The Tweet CDC dataset was created by annotating a collection of 52 million tweets for CDC. This collection of tweets was taken from the 1% sample of all tweets posted between 30th June 2011 and 15th September 2011 then filtered for English tweets (Lui and Baldwin, 2012).

Mentions were extracted using the state of the art NER system available at the time of dataset construction (Ritter et al., 2011). This resulted in a stream of 6 million mentions. Approximately 10% of all tweets contain a person mention.

Two common first names were chosen for annotation; common names were chosen by counting the frequency of the names in a gazetteer and then choosing uniformly at random a low frequency name (1,000 - 10,000) and a medium frequency name (10,000 - 100,000). The low frequency name 'Roger' and medium frequency name 'Jessica' were chosen. Table 3.2 details some statistics about the dataset.

Name	Mentions	Entities	Wiki Exists
Roger	5,794	137	69%
Jessica	10,543	129	46%
Roger + Jessica	16,337	266	58%

Table 3.2: Amount of mentions/entities per mention texts and the percenage of the entities that have a Wikipedia page about them.

Table 3.2 shows that a large proportion of the entities mentioned on Twitter do not have a Wikipedia page. As only popular entities have Wikipedia pages, any significant popularity bias has been avoided. Table 3.3 lists a representative sample of entities and their description highlighting the broad range of entities in the dataset:

Entity	**Description**
Roger Federer	Popular Tennis Player
Jessie J	Popular musician whose real name is ‘Jessica Cornish’
Jessica Rowling	Daughter of popular author J.K Rowling
Roger Alton	Newspaper Executive
Jessica Hess	Painter
Jessica Zelinske	Playboy ‘Babe’
Roger Hill	Weatherman
Roger	Character from popular TV show ‘American Dad’
Jessica Drew	aka Spider-Woman

Table 3.3: Information about some of the entities included in the Tweet CDC dataset.

For any CDC dataset, the mention frequency (how often an entity is mentioned) distribution should show a power-law distribution; most entities are mentioned infrequently, and there are a small number of entities that are mentioned very frequently. Figure 3.2 shows the mention frequency distribution for this dataset, clearly showing a power-law distribution.

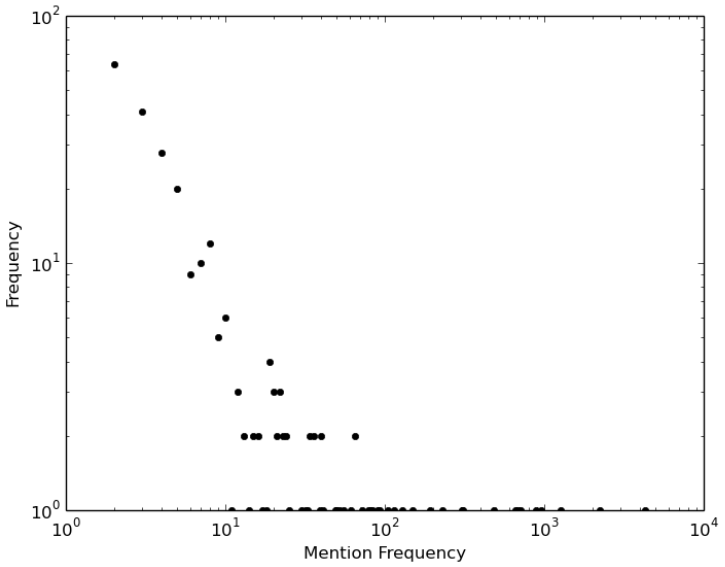

Figure 3.2: Mention frequency distribution for the TweetCDC dataset.

Typically, the more frequently an entity is mentioned, the more unique mention texts are used to refer to it. This is partially due to social pressures to create a unique

way to reference a common entity and the increased likelihood of spelling mistakes. Figure 3.3 shows number of unique mention texts against mention frequency, demonstrating a clear positive correlation.

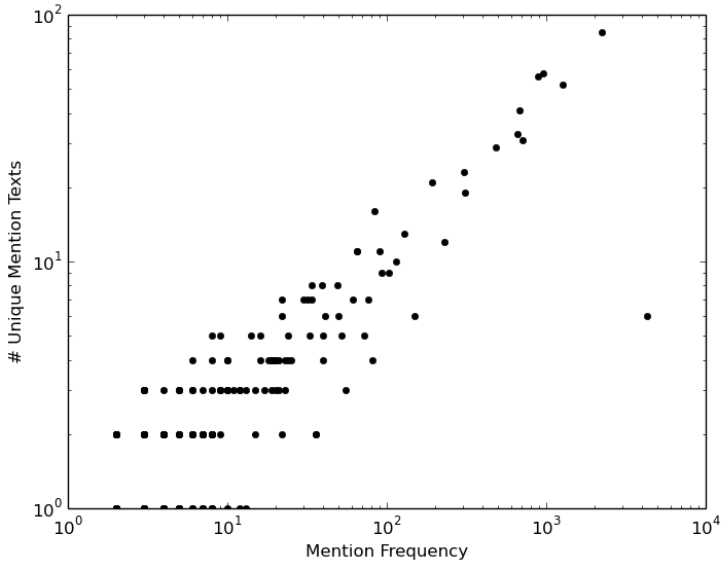

Figure 3.3: Amount of unique mention texts against frequency for each entity in the TweetCDC dataset.

As outlined in Figure 3.1, the dataset should contain both polysemy and synonym ambiguity; this can be demonstrated by computing some simple statistics:

- 32% of entities are referred to using a single unique mention text. These entities are trivial to resolve.
- 60% of entities are referred to using more than one mention text (synonymy ambiguity).
- 22% of entities are referred to using a single but not unique mention text.
- 46% of entities are referred to using a mention text that is not unique to the entity. (polysemy ambiguity).

A more detailed analysis of the ambiguities in the dataset can be performed by estimating the amount of polysemy/synonymy ambiguity for each entity:

- **Synonymy Ambiguity Estimate:** The more an entities mentions are spread out over multiple mention texts, the harder an entity is to resolve due to synonymy.

To estimate synonymy ambiguity, the proportion of mentions using each mention text is computed. One minus the average is reported so no ambiguity is mapped to zero.

- **Polysemy Ambiguity Estimate:** The more the mention texts used to refer to an entity are used to refer to a different entity the more difficult it is to resolve due to polysemy ambiguity. A simple estimate of polysomy ambiguity is the average probability each mentions mention text refers to a different entity.

Plotting these estimates for each entity in Figure 3.4 clearly shows these ambiguities exist in the dataset. This Figure highlights some interesting features of the dataset:

1. The bottom left represents entities where a single, unambiguous mention text is used to refer to the entity.
2. Most entities are near the axis; this demonstrates that typically entities exhibit a single type of ambiguity.
3. There are entities with both polysomy and synonymy ambiguity.
4. No entities exist near the top right where many mention texts are used to refer to the entity and those mention texts are often used to refer to other entities.

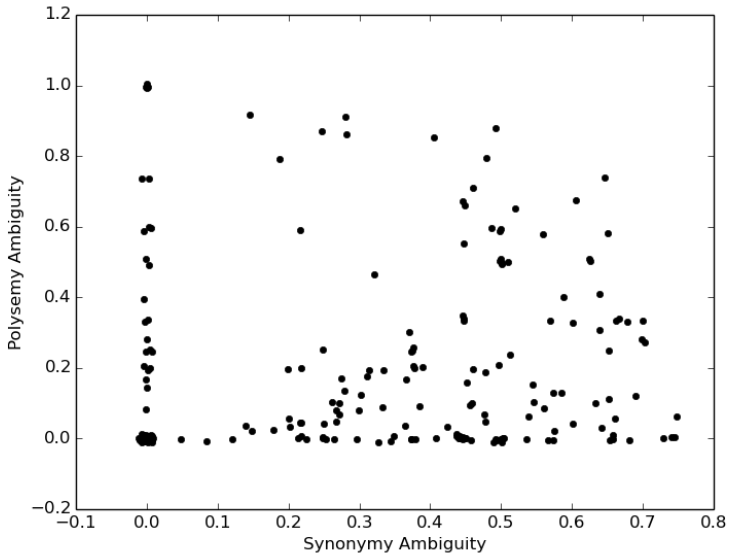

Figure 3.4: Amount of unique mention texts against frequency for each entity in the TweetCDC dataset.

3.2 Annotation Verification

Ideally when verifying the annotations are correct a second annotator would annotate a portion of the dataset. Due to the amount of effort required to create the dataset this is infeasible. Various experiments were performed to identify a technique that other annotators could use to efficiently annotate a sample of the data. Techniques to sub-sample the data resulted in datasets either extremely biased (only multi term names) or trivial (reporting incorrectly perfect inter annotator agreement). Annotators also struggled to pick up the necessary background knowledge about entities being annotated, frequently incorrectly reporting mistakes[1].

3.3 Evaluation

CDC is difficult to evaluate, there is no intrinsic value in grouping all the mentions by the entity being referred to. A measure of 'usefulness' needs to be approximated (Von Luxburg et al., 2012). This could come from a downstream task such as question answering. If a CDC system improves a question answering system it has value, it helps people find out the correct answers to questions. Unfortunately evaluating CDC with a downstream task has not been done and is beyond the scope of this thesis. In this section existing CDC measures that approximate usefulness as 'similarity to gold standard' are considered.

The standard practice for evaluating CDC is to use WDC evaluation measures. This presents two problems: None of these evaluation measures were designed for CDC and they are not directly applicable for use with partially annotated datasets.

While WDC and CDC are similar tasks, there are important differences with the two problems: The mention frequency distribution and the types of mentions being resolved. These differences can cause problems for existing evaluation measures.

In Section 3.1, an efficient technique for creating CDC datasets was introduced, achieved by annotating only a sample of entities (partial annotations). Existing evaluation measures require all (or all non-singleton) entities to be annotated. In this section, a simple modification is proposed and experiments demonstrate the new evaluation measure is faithful to the original measure.

[1] This is an accepted limitation of this thesis

3.3.1 Existing Evaluation Measures

One significant problem with WDC is the lack of a well-justified and agreed-upon evaluation measure. Since the 2011 CoNLL shared task on WDC (Pradhan et al., 2011), the standard has been to evaluate using all of MUC, B^3 and CEAFe (described below) taking the average F-score as the primary evaluation measure.

The annotations provide sets of mentions that refer to the same entity, referred to as *key entities* (K). The system outputs sets of mentions that it predicts refer to the same entity, the *response entities* (R).

The WDC evaluation measures were originally designed for use with gold standard mention extraction (both key and response entities cover the same mentions), and alternative measures were created to take into account spurious (incorrectly predicted mentions) and missing mentions (Stoyanov et al., 2009; Rahman and Ng, 2009; Cai and Strube, 2010). These were found to be redundant, so simpler modifications were proposed that are more faithful to the original evaluation measures (Pradhan et al., 2014). These versions of the evaluation measures are described below.

3.3.1.1 MUC

MUC (Vilain et al., 1995) is a link-based evaluation measure that models entities as minimum spanning trees (MST) over the mentions in each entity. These models count the mistaken branches to evaluate precision and the missing branches to evaluate recall. It lacks a clear and intuitive mathematical definition though it can be (semi-formally) summarised as:

Precision = 1 - Number of MST links mistakenly predicted / Amount of MST links predicted

Recall = 1 - Number of MST links required to unite key entities / Amount of MST links in the key entities.

3.3.1.2 B^3

B^3 (Bagga and Baldwin, 1998a) is a mention-based evaluation measure that was designed to overcome two (arguably) undesirable properties of MUC:

1. A single MST link can be responsible for multiple errors. Merging two completely correct response entities is a single error according to MUC irrespective of the amount of mentions in each entity.

2. As MUC is link-based it is unable to evaluate singletons.

For each mention (m_i), the precision and recall is computed using equation 3.1. K_{m_i} is the key entity that contains mention m_i and R_{m_i} is the same for the response entities. The final precision is the average over all response mentions whereas recall is averaged over key mentions.

$$p = \frac{|R_{m_i} \cap K_{m_i}|}{|R_{m_i}|} r = \frac{|R_{m_i} \cap K_{m_i}|}{|K_{m_i}|} \tag{3.1}$$

3.3.1.3 CEAFe

CEAFe (Luo, 2005) is an entity-based evaluation measure. Each response entity should represent a unique key entity. This is not enforced in any of the previous evaluation measures where multiple key entities may contribute to evaluating a single response entity. This can lead to trivial responses (for example: all singletons) scoring surprisingly well.

CEAFe computes the optimal alignment between the key and response entities using equation 3.2 to measure entity similarity. Precision and recall is computed using equation 3.3 where Φ^* is the sum of the entity similarities for the optimal alignment.

$$\Phi(X,Y) = \frac{2|X \cap Y|}{|X| + |Y|} \tag{3.2}$$

$$p = \frac{\Phi^*}{|R|} r = \frac{\Phi^*}{|K|} \tag{3.3}$$

An important design feature of CEAFe was interpretability. It represents the 'average correctness of the entities'. It is very clear if a system is producing too many/few response entities by comparing the precision and recall values; a larger precision implies too many response entities and vice versa.

3.3.2 Evaluating CDC with WDC Evaluation Measures

There are important differences between WDC and CDC that can influence the evaluation measure.

3.3.2.1 Mention Frequency

Typically, WDC corpora contain a collection of newswire documents; each document normally has a couple of hundred mentions. With WDC, the evaluation is over multi-

ple documents where each document contributes its own 'small' WDC problem where the maximum mention frequency is bounded by the amount of mentions in a document. CDC is evaluated over a single document collection, and this collection can be any size with no maximum mention frequency. Due to the power law mention frequency distribution, a small number of frequently mentioned entities can dominate the dataset/evaluation. This is clearly demonstrated by the relative size of the most frequently mentioned entities from a WDC dataset OntoNotes (Weischedel et al., 2011) (used for the CoNLL 2011/2012 shared tasks) and the annotated mentions from the TweetCDC dataset as shown in Figure 3.5.

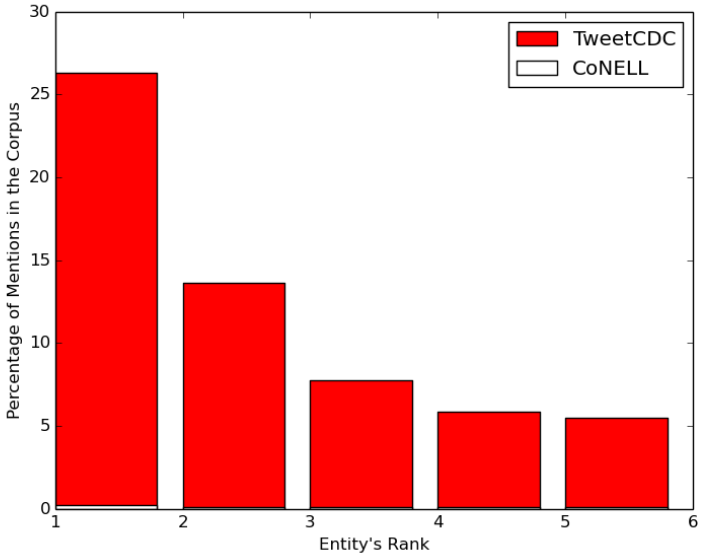

Figure 3.5: The five most frequently mentioned entities and their mention frequency relative to dataset size.

When using MUC or B^3, the more frequently an entity is mentioned, the more it contributes to the evaluation. The frequently mentioned entities make a disproportionate contribution not envisaged when the original evaluation measures were proposed. With CEAFe, each entity makes the same contribution to the final evaluation score irrespective of its mention frequency.

3.3.3 Mention Types

With most CDC systems, only proper noun references are resolved; for most entities, its most frequent mention text is used in a large proportion of its mentions (77% for the

average entity annotated in the TweetCDC dataset). This introduces a lot of mentions that are trivial to resolve.

WDC involves resolving multiple types of mentions (proper noun, common noun and pronoun mentions). There are a small number of cases where resolution is trivial (for example: apposition and exact proper noun match) resolving most mentions requires combining imprecise syntactic and semantic information.

The large amount of trivial mentions in a typical CDC corpora were not considered when the WDC evaluation measures were originally created. The large amount of trivial mentions can dominate all of MUC, B^3 and CEAFe potentially reduce the evaluation measures sensitivity.

3.3.4 Evaluating with Partial Annotations

With the TweetCDC dataset introduced in section 3.1, a sample of the entities were annotated, providing partial annotations. Existing evaluation measures were designed to deal with spurious mentions (incorrectly identified mentions). Spurious mentions appear similar to unannotated mentions (both are not in any key entity) though there is an important difference: With spurious mentions, their relationship to all other mentions is known (not coreferent to any other mention) whereas, with unannotated mentions, their relationship is unknown, so it is possible for two unannotated mentions to be referring to the same entity. Figure 3.6 demonstrates this problem.

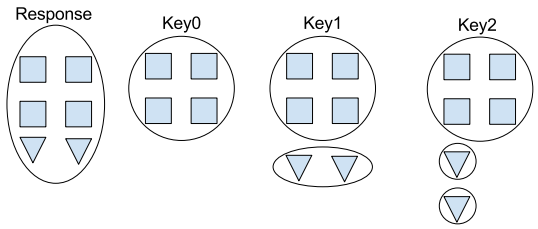

Figure 3.6: An example response containing spurious/unannotated mentions (triangles) with three possible sets of key entities.

Figure 3.6 shows a response that include some spurious/unannotated mentions and three possible sets of key entities. If they are spurious mentions none of them refer to each other, so key0 is valid. If they are unannotated mentions, either key1 or key2 could be the truth as the coreference relationship between the unannotated mentions is unknown.

Unannotated mentions cause a problem for MUC and B^3 as they require the coreference relationships between all mentions in each response entity to be defined. With CEAFe, only the similarity between the key and response entities is required.

CEAFe is clearly the most appropriate evaluation measure when evaluating CDC on a dataset with partial annotations. A naive application of any WDC evaluation measure is clearly not representative. It evaluates how well a small set of key entities is represented by a much larger set of response entities. A simple adjustment of the response entities by removing any entity that contains no annotated mentions means that the key entities are being evaluated against the set of response entities that could be used to represent the key entities.

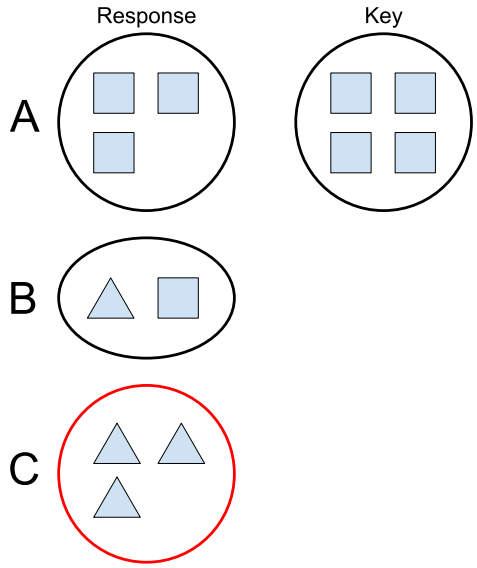

Figure 3.7: An example to show which entities are removed prior to evaluation. Annotated mentions are squares, unannotated mentions are triangles and the response entity in red is removed.

The adjustment to CEAFe is demonstrated in figure 3.7. Response entity C is removed prior to evaluating using the original CEAFe. Response entities A and B demonstrate that the key entity is almost perfectly constructed with one mention being incorrectly grouped with an unannotated mention. The key entity is spread across two response entities. Response entity C provides no information about how well the key entity was represented in the response. If response entity C is not removed CEAFe considers it a 'completely incorrect' response entity and evaluates the response as if the key entity was spread over three response entities.

This adjustment to CEAFe is the same as normal CEAFe when all key and response mentions are annotated. It is used thought the remainder of this thesis.

3.3.5 Experiments

Experiments to verify the discussion above and demonstrate that CEAFe is robust and capable of evaluating datasets with partial annotations were performed using the TweetCDC dataset.

3.3.5.1 Dataset/System

The TweetCDC dataset is reduced to the annotated mentions only (creating the Annotated-TweetCDC dataset). This is representative of a typical CDC dataset where only annotated mentions are considered. With this dataset, key entities can be removed to simulate evaluating using a partially annotated dataset.

Two baseline systems, designed to have a clear performance difference, were implemented. Both systems used average link hierarchical agglomerative clustering (a common model to use as outlined in Section 2.4) The only difference between the two systems is the mention similarity measure, both of which use standard techniques as outlined in Section 2.

1. **Mention Only (mt):** Mention similarity was measured using cosine similarity of character bigram indicator vectors.

2. **Mention and Context (mt_ct):** Mention similarity was measured using a linear combination of mention text similarity and contextual similarity weighted 0.8 and 0.2 respectively (Rao et al., 2010; Singh et al., 2011). Mention similarity was measured using the same technique as the mention text only system. Contextual similarity was measured using cosine similarity of tf-idf weighted document (tweet) terms.

Clearly, the system that uses both contextual and mention similarity should perform better than the mention text only system.

3.3.6 Comparing Systems

A response (system output) is produced for each of the two systems. They each require a single parameter (the threshold for the clustering to stop) to be set. The parameter

was chosen so the response is reasonable (but not optimal) performance for all the evaluation measures without using an evaluation measure to set the threshold. This was achieved by setting the parameter such that the same amount of response and key entities were produced. This response is used for all experiments (the parameter is only set once).

Table 3.4 reports the performance of the two systems when no key entities were removed. The MUC and B^3 scores for the two systems are very similar demonstrating their lack of sensitivity. The MUC scores are very high; given that the systems implemented are baselines, the evaluation measure should ideally not report performance that high. There is clearly a performance difference with CEAFe.

	MUC			B^3			CEAFe		
System	P	R	F	P	R	F	P	R	F
mt	98.0	98.0	98.0	77.9	86.4	81.9	44.7	44.7	44.7
mt_ct	98.2	98.2	98.2	78.3	86.7	82.3	49.2	49.2	49.2

Table 3.4: Results from evaluating the mt and mt_ct systems on the Annotated-TweetCDC dataset.

3.3.6.1 Evaluating with Partial Annotations

To demonstrate the problem with treating unannotated mentions as spurious mentions, a proportion of the key entities were removed uniformly at random, simulating partial annotations. The representative response was evaluated against 10,000 different sets of key entities for various proportions of removed key entities. The only modification to the evaluation measures made was removing all response entities that contained no annotated mentions.

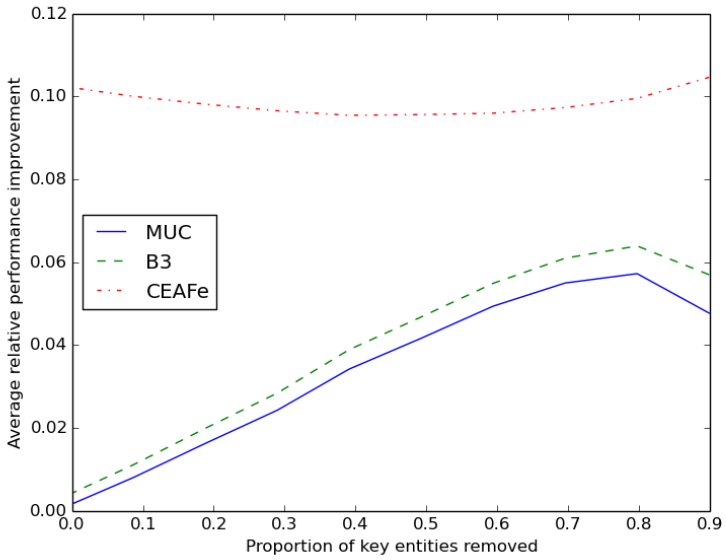

Figure 3.8: The average performance improvement observed when a proportion of the entities are removed from the Annotated-TweetCDC dataset.

Figure 3.8 shows the average performance improvement between the two systems for all of the evaluation measures. It is clear that the presence of unannotated mentions in the response affects MUC and B^3, but not CEAFe. As discussed previously, MUC and B^3 require all the coreference relationships in a response entity to be defined whereas CEAFe does not.

3.4 Summary

In this chapter, existing techniques to create a CDC dataset were discussed, demonstrating that manual annotation is the only viable technique to produce a CDC dataset.

To create a streaming CDC dataset, a new technique was introduced: Instead of annotating all mentions, only a sample of entities were annotated (partial annotations). This provides an efficient technique to create a large streaming CDC corpus with few compromises.

Typically, WDC evaluation measures have been applied to CDC without any discussion of their appropriateness. In this chapter, the differences between WDC and CDC and the effect on the evaluation measures were discussed, demonstrating problems with MUC and B^3.

Existing evaluation measures are not directly applicable to CDC datasets with par-

tial annotations. The appropriateness of existing evaluation measures were discussed with CEAFe being clearly the most appropriate evaluation measure. A small adaptation was proposed in Section 3.3.4 to ensure that the evaluation is true to the design goals of the original evaluation measure.

Chapter 4

Sampling Techniques for Streaming CDC

To process an infinite stream of mentions, a CDC system must satisfy the constraints of the streaming model of computation (outlined in Section 2.4.1): it must use a constant amount of memory and a constant amount of time per mention.

Only a small portion of the previously seen information can be stored. Choosing what to store is difficult, there is no general solution. Each task has a unique set of challenges that should be addressed when deciding how best to use the available memory.

In this chapter, a sampling approach to streaming CDC is introduced. The challenges faced by a streaming CDC system are described and new sampling techniques that address these specific challenges are introduced. These new techniques significantly improve upon existing techniques, achieving 95% of the performance of a non-streaming system while using 80% of the memory required for a non-streaming system.

4.1 Sampling

Sampling the process of selecting items from a population, is widely used. The classical use case is efficient estimation. With opinion polling or population counting, it is not feasible to ask/count everyone. Instead, a sample of people/areas can be queried (where the sample size is determined by a 'budget') and results used to estimate population statistics.

Sampling techniques are widely used to make infeasible tasks possible. Such as creating datasets for machine learning tasks. When creating an annotated dataset from

a large corpus, annotating all items is infeasible; instead, a (typically uniform) sample of items is annotated. If the dataset is created to train a machine learning model, the annotation budget is best spent on the items that are most useful to the model. Instead of a uniform sample, active learning (Tong and Koller, 2001) can be used to sample items that will be the most useful to the model. The usefulness of each item is predicted given an initial model; then, the most useful items are annotated and the model is re-trained.

Numerical optimisation can be viewed as a sampling task. At each step, the goal is to sample a better solution. The maximum sample size is the maximum amount of iterations. Grid search performs uniform sampling whereas techniques such as gradient descent use a model to sample points that are likely to satisfy the goal, reducing the amount of samples required.

Sampling provides a convenient way to make complex tasks feasible. While basic (typically uniform) sampling techniques can be effective, using a sampling technique that addresses the task goals can result in a significant performance improvement. This chapter identifies the challenges streaming CDC systems must overcome and demonstrates how addressing these challenges with new sampling techniques results in a significant performance improvement.

4.2 Sampling from Streams

The constraints of the streaming model of computation (as outlined in Section 2.4.1) introduces a limit on the amount of memory available. Without a formal limit, a system that stores the stream (or a proportion of it) will eventually run out memory and crash; this necessitates sampling. The sampling technique must maintain a constant sized set of items. Various sampling techniques have been applied as parts of existing (non CDC) systems.

Streaming first story detection (Petrović et al., 2010) involves identifying documents that mention a new event. Incoming documents are compared to a sample of previously seen documents and a novelty score is assigned. News stories typically break and then gradually die out, so only the most recent documents are required leading to the moving window technique used.

Streaming language modelling (Osborne et al., 2014) involves sampling previously encountered documents to build the best possible language model for the current documents. The main challenge is how to avoid over-representing bursting information.

By maintaining a sample exponentially biased against age (consequently not all items are added to the sample), new information is not over-represented and bursts do not adversely affect the language model.

The best sampling technique to use depends on the task, for example: If recency is the only important aspect, moving windows (storing the k most recent items) work well though if older information is required then more advanced techniques should be used.

4.3 Streaming CDC Challenges

Streams such as Twitter are known to be noisy (entities are referred to in many different ways), real time (the set of entities being referred to changes constantly), highly bursty (some entities will suddenly become very popular) with a broad coverage (many topics/entities are being discussed). The temporal properties are particularly challenging for streaming CDC. If a system samples a moving window of the most recent mentions, it will model the real-time nature of the stream though when a bursting entity is encountered, it will dominate the sample and the system will be unable to resolve the broad range of entities being mentioned. There are a number of challenges that must be addressed within the constraints of the streaming model of computation:

1. **Recency:** With real-time streams, the topics being discussed are constantly changing. This means the most recent information is likely to be the most useful to help resolve a mention. For example: Mentions of British Olympian 'Jessica Ennis' during the 2012 Olympics is less useful at the 2016 Olympics as she changed her name to 'Jessica Ennis-Hill'. A streaming CDC system should retain recent information.

2. **Distant Reference:** As a broad range of entities are discussed, some will be mentioned infrequently, with large gaps between their mentions. For example: people who play in sports teams may only be mentioned at the weekends. Resolving these mentions is particularly challenging due to the necessity to forget information. Distant reference is the opposite of recency. A CDC sampling technique will need to balance the amount of recent information stored against the amount of older information stored to resolve both distant and recent references.

3. **Entity Diversity:** A broad range of entities are constantly being discussed on Twitter, even when a single entity suddenly becomes very popular (a bursting

entity, for example: 'Phillip Hughes', a cricketer who died following a bowling injury). For an entity to be correctly resolved, it must be represented in the sample (if it was previously encountered). A CDC sampling technique should maximise the amount of entities represented in the constant amount of memory available.

These factors are clearly shown in the TweetCDC dataset. If recency is important, the distribution of time between successive mentions will be biased towards zero (there is typically a short time between mentions of the same entity). If distant reference is also important, the distribution will decay gradually (there are sometimes long gaps between mentions of the same entity) as shown in Figure 4.1.

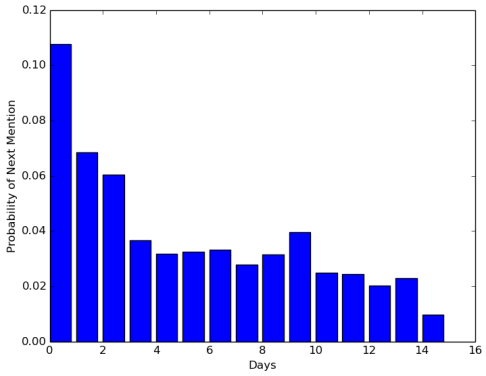

Figure 4.1: Distribution of time between successive mentions in days in the TweetCDC dataset.

To demonstrate the importance of maintaining a diverse set of entities, the impact of bursting entities on a system that only models recency is shown. A moving window of 1 day was maintained and the proportion of mentions in the window that refer to each entity was calculated every hour. This produces a smoothed mention rates as shown in Figure 4.2. It is clear that, when some entities burst, they take up a significant portion of the window leaving less space for other entities to be represented, reducing the diversity of the sample.

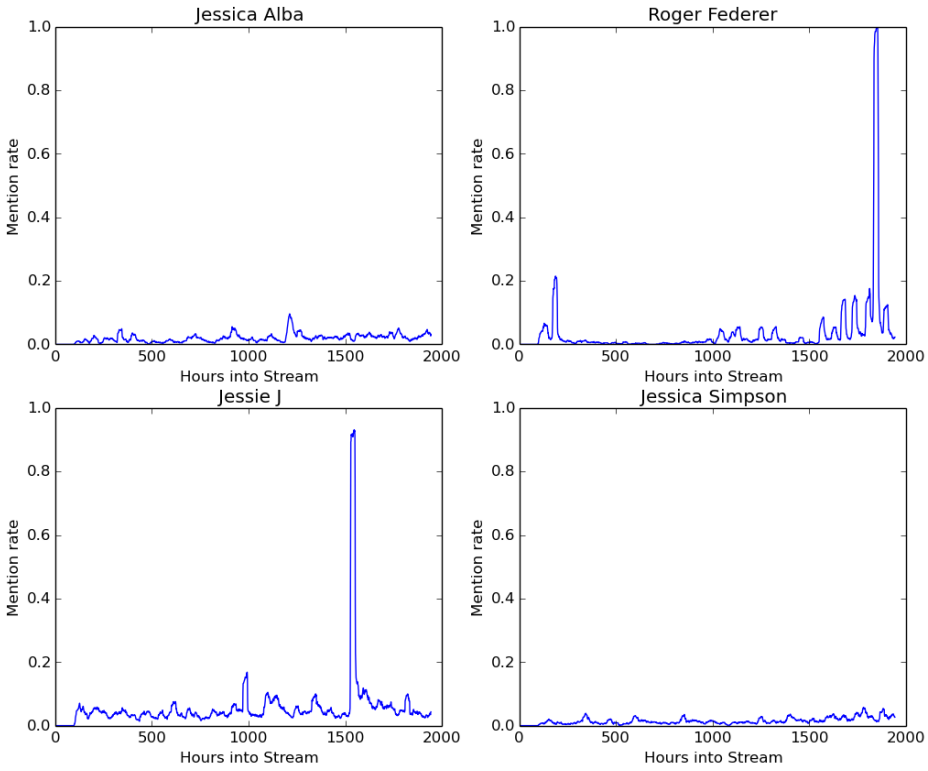

Figure 4.2: Mention rate for the four most frequently mentioned entities in the TweetCDC dataset.

4.4 A Streaming CDC system

Before the new sampling techniques are introduced, a system for streaming CDC resolution is presented. A streaming CDC system must process mentions sequentially, providing an instant response in a constant amount of time while using a constant amount of memory.

A comparable non-streaming system will still process mentions sequentially though without any time/memory constraints. A batch system that accesses the mentions in any order has a significant advantage. It is able to avoid making retrospectively 'bad' decisions that are made due to processing the stream sequentially.

All CDC systems that process mentions sequentially will have the same input and output. The input is a stream of mentions and the output is a stream of entity identifies where all mentions with the same identifier are predicted to refer to the same entity. As each mention is processed, the system must output its (the current mentions) entity

identifier before progressing onto the next mention as shown in Figure 4.3.

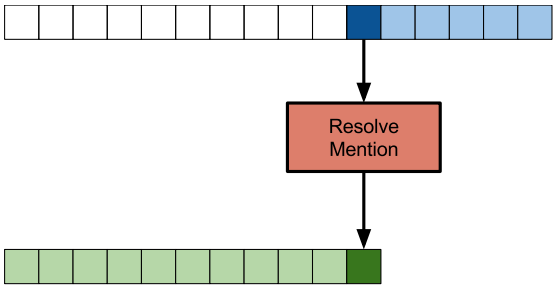

Figure 4.3: Diagram representing a system processing a stream of mentions (in blue) and outputting entity identifiers (in green).

A sequential pairwise single link agglomerative clustering system was implemented due to its efficiency, and it can easily be adapted to the streaming model of computation. The current mention is compared to previously seen mentions, identifying its nearest neighbour. If this similarity is above a threshold (the linking threshold), the system outputs the same entity identifier; else a new entity identifier is output. This system (as outlined in Figure 4.4) processes mentions sequentially but does not conform to the streaming model of computation as all previously encountered mentions are stored.

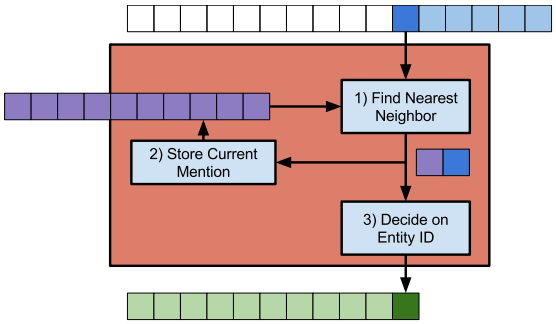

Figure 4.4: Diagram showing mentions being resolved by a system that does not conform to the streaming model of computation, storing all the previously encountered mentions.

To adapt this system to the streaming model of computation instead of storing all previously seen mentions, a sample of the mentions are stored as shown in Figure 4.5. The constant memory constraint is satisfied by the constant size of the sample and the constant time constraint is satisfied as it takes a constant time to compare the current mention with a constant amount of mentions. The sampling technique should address the streaming CDC challenges outlined in Section 4.3.

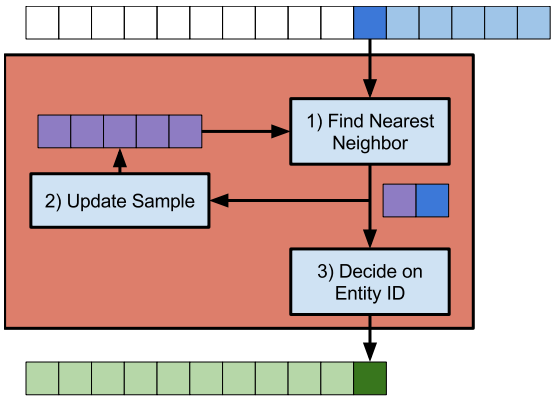

Figure 4.5: Diagram showing mentions being resolved by a system that conforms to the streaming model of computation where a sample of previously encountered mentions are stored.

Mention similarity is measured using a standard mention similarity measure from the CDC literature (Rao et al., 2010; Singh et al., 2011). It takes a linear combination of mention text similarity and contextual similarity shown below with their weights (the same weights were used in previous work) shown in brackets:

- Mention Text Similarity (0.8) : Cosine similarity of character bigram indicator vectors[1] is an effective and efficient way to measure mention text similarity.
- Contextual Similarity (0.2) : Cosine similarity of tf-idf weighted document terms. This is the standard document similarity measure used in many domains.

4.5 Problems with Existing Sampling Techniques for CDC

Existing sampling techniques are described and the streaming CDC challenges they address are identified. Figure 4.6 shows the running example that will be used to demonstrate each technique. There is an artificial stream of 27 mentions: all entities with more than one mention (non-singleton) are shown with coloured diamonds indicating the entity being mentioned. Mentions 13 and 23 will be used to demonstrate the techniques. They are particularly challenging to resolve as there are large gaps between them and their previous mentions. Just before mention 13, there is a small burst

[1]Bigrams were determined through experiments on the development portion of the dataset

of mentions for a different entity demonstrating the problems that can be encountered with bursting entities.

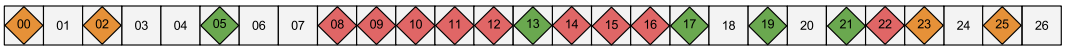

Figure 4.6: The running example used to demonstrate the various sampling techniques.

4.5.1 Sequential System (No sampling)

To provide an upper bound on system performance, a system that performs no sampling, instead storing all previously seen mentions is used. Figure 4.7 shows the running example where all previous mentions are stored allowing mentions 13 and 23 to be correctly resolved. Psudocode can be found in Appendix A.1.

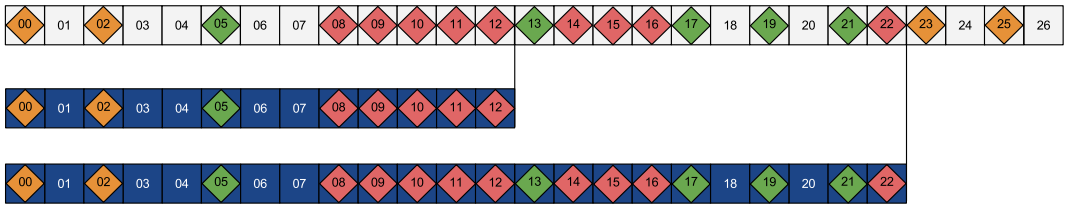

Figure 4.7: Running example demonstrating the sequential (no sampling) technique.

4.5.2 Window Sampling

The simplest sampling technique is to only store the most recent mentions. This only models recency. In the running example shown in Figure 4.8, mentions 13 and 23 will be incorrectly resolved as none of their previous mentions are in the sample. The sample just before processing mention 13 is dominated by mentions of the red entity due to its burst. Psudocode can be found in Appendix A.2.

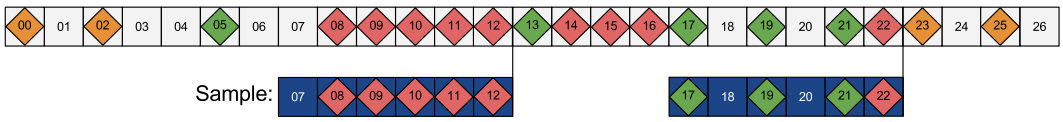

Figure 4.8: Running example demonstrating the window sampling technique.

4.5.3 Reservoir Sampling Techniques

Reservoir sampling approaches are standard techniques from the streaming algorithms literature. Unlike traditional (batch) sampling techniques that produce a static sample, these techniques constantly update the sample as the stream is processed, maintaining a sample that has the desired properties.

Reservoir sampling techniques provide a convenient way to model the recency/distant reference trade-off. By reducing the amount of mentions that enter the sample older mentions can remain in the sample allowing distant references to be resolved.

4.5.3.1 Biased Reservoir Sampling (Aggarwal, 2006)

Window sampling maintains a deterministic window; a mention is removed after k mentions are seen. Biased reservoir sampling maintains a probabilistic window where the probability a mention remains in the sample decays exponentially with age. This randomisation allows some mentions older than the window size to remain in the sample which may help resolve distant references.

The sample is maintained by inserting items with a constant probability (P_i). When an item is inserted, it replaces an item uniformly at random. For a sample of size k, after n removal operations, the probability an item remains in the sample is $(1-\frac{1}{k})^n$ though as an item is only removed when one is inserted, after inserting the item then processing m mentions the probability an item remains in the sample is $P_i(1-\frac{P_i}{k})^m$. The probability an item remains in the sample decays exponentially with age and the insertion probability (P_i) controls the steepness of the decay as shown in Figure 4.9.

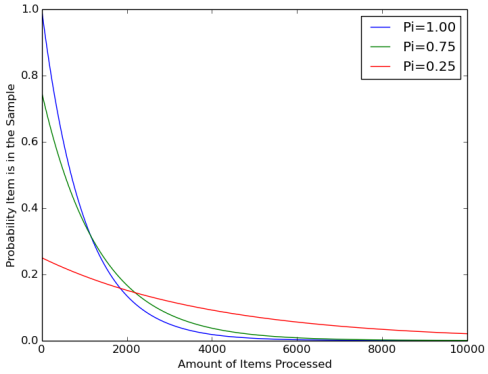

Figure 4.9: Probability an item remains in the sample for various insertion probabilities. Sample size = 1000.

By lowering the insertion probability, fewer mentions are inserted into the sample decreasing how well recency is modelled. Mentions in the sample may stay in longer, potentially helping resolve distant references. When decreasing the insertion probability, fewer mentions enter the sample which limits how well distant reference is modelled for infrequently mentioned entities as they are unlikely to ever be represented in the sample.

Figure 4.10 shows the running example with biased reservoir sampling. It is possible for mention 13 to be correctly resolved (unlike with window sampling) as there is a chance mention 5 will remain in the sample. Psudocode can be found in Appendix A.3.

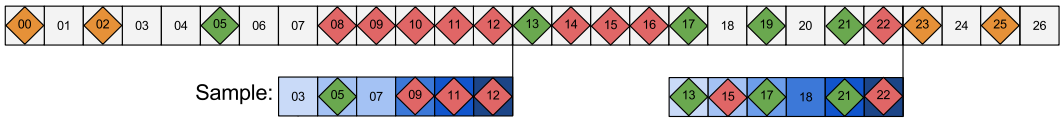

Figure 4.10: Running example demonstrating the exponential reservoir sampling technique. Probability a mention is in the sample is indicated by background shade, lighter is lower probability.

4.5.3.2 Uniform Reservoir Sampling (Vitter, 1985)

Uniform reservoir sampling maintains a uniform sample over all previously seen items.

When processing the n-th item in the stream the probability each previous item is in the sample should be $\frac{k}{n}$. The probability an item remains in the sample should decay exponentially as the stream is processed. Adding an item with probability $\frac{k}{n}$, replacing an item uniformly at random ensures items enter the sample with the correct probability and the probability they remain in the sample decays exponentially resulting in a uniform sample.

This technique does not model recency; instead, it samples a temporally diverse set of mentions that may help resolve distant references. As very few mentions ever enter the sample, only a small range of entities will ever be represented, and bursting entities are unlikely to be over-represented in the sample.

Figure 4.11 shows the running example with uniform reservoir sampling. While it is possible for mentions 13 and 23 to be correctly resolved, it is unlikely a mention of the same entity will be in the sample. Psudocode can be found in Appendix A.4.

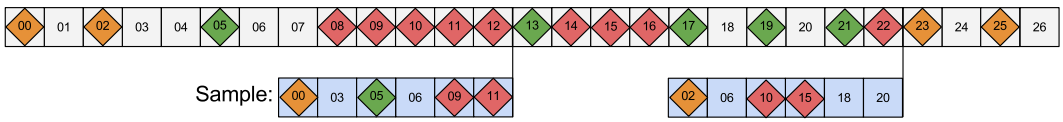

Figure 4.11: Running example demonstrating the uniform reservoir sampling technique. Probability a mention is in the sample is indicated by background shade, lighter is lower probability.

4.6 Improved Sampling Techniques

In this section, new sampling techniques are presented. They improve upon the existing techniques discussed.

4.6.1 The Recency vs Distant Reference Trade-off

To address the recency/distant reference challenge a sampling technique should be able to trade-off storing fewer recent mentions in exchange for storing mentions that are likely to be required for resolving distant references.

Biased reservoir sampling can trade off how well recency is modelled in exchange for storing older mentions allowing distant references to be resolved. This trade-off involves some mentions never entering the sample, a very severe way to reduce how well recency is modelled. Some entities with few mentions will never enter the sample and can never be correctly resolved. The recent past is only fully modelled if all mentions enter the sample, either with window sampling or biased reservoir sampling where $P_i = 1$.

To resolve distant references, mentions that are likely to be useful in the future should not be removed from the sample. To implement this, when a mention is identified as 'likely to be useful in the future', it enters a first-in-last-out cache; while it is in the cache, it will not be removed from the sample.

It would be possible to produce a complex model for predicting a mention's future usefulness although, to reduce any computational overhead and maintain within the constraints of the streaming model of computation, a simple model is used. If a mention is identified as a nearest neighbour, it is likely to be useful again in the future and enters the cache. This simple model requires no computational overhead given the system outlined in Section 4.4.

Figure 4.12 outlines the cache sampling technique (psudocode can be found in Appendix A.5). The size of the cache determines how much emphasis is put on distant reference. The remainder of the sample should model recency. There are two techniques that model recency: window sampling (Cache-Win) and biased reservoir sampling where $P_i = 1$ (Cache-Exp).

Cache sampling introduces a parameter, the size of the cache. The size of the cache is the percentage of mentions in the sample that can be in the cache. The cache can be thought of as a flag on the mentions in the sample indicating their position in the cache rather than a separate pool of memory.

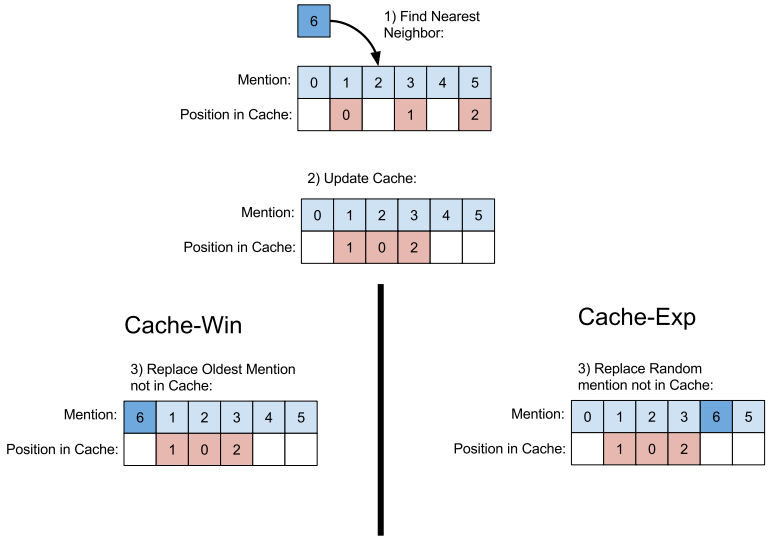

Figure 4.12: Diagram of the cache sampling technique.

Figure 4.13 shows the running example with these sampling techniques. The example assumes the system perfectly resolves all the mentions so mention 0 enters the cache (purple background) when mention 2 is processed as mention 0 is identified as its nearest neighbor. Mention 8 stays in the cache as it is identified as the nearest neighbor for mentions 9, 10, 11, 12. Mention 05 never enters the cache as it is never identified as a nearest neighbor. Cache sampling did not help resolve mentions 13 and 23 as there was too much of a gap. Mentions 8 and 13 are in the cache at the end having been correctly identified as useful and stayed in the sample for all remaining mentions of their entity.

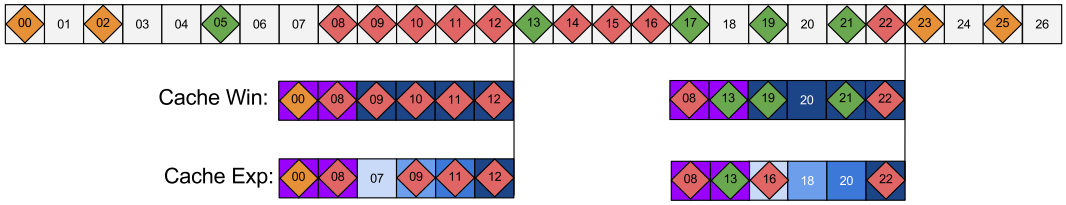

Figure 4.13: Running example demonstrating both the cache sampling techniques. Probability a mention is in the sample is indicated by background shade, lighter is lower probability. The purple background represents mentions in the cache.

4.6.2 Modelling Entity Diversity

The cache sampling techniques deal with the recency/distant reference trade-off, they fail to help ensure entity diversity in the sample. Biased reservoir sampling can reduce the impact of bursting entities by reducing the amount of mentions that enter the sample, at the expense of modelling recency. Uniform reservoir sampling ensures bursting entities do not impact the sample by maintaining a temporally diverse set of mentions although it makes no attempt to model recency. Neither technique provides a way to reduce the impact of bursting entities without impacting other factors.

Reducing the amount of mentions stored for each entity will reduce the impact of bursting entities and allow a diverse range of entities to be represented in the sample. If the current mention is sufficiently similar to an existing mention in the sample, it will update the existing mention instead of replacing a mention. The current mention should only replace existing mentions if it is sufficiently novel.

When an entity bursts many similar mentions are created, typically related to the reason why the entity started bursting. For example Figure 4.14 shows tweets that mention ‘Jessie J’ after her VMA (MTV Video Music Awards) performance where most mention texts were identical and the tweets include the term ‘VMA’. Storing fewer of these mentions will limit the impact of a bursting entity.

```
Photo:  Selena Gomez talking to Jessie J at the VMAs !
If I ever broke my leg I want my cast dazzled out like Jessie Js on the VMAs
#VMAs Jessie J is doing an a brilliant job, singing Rainbow now, on her throne.
Pitbull, Nayer and Ne-Yo just performed Give Me Everything..
Jessie J Talks 'Domino' Dedicates Her VMA Performance To The Heartbeats
Jessie J talks about VMA performance style .  http://t.co/HYxgGC8
```

Figure 4.14: Tweets about Jessie J sent after her VMA performance, mention text is coloured red.

Even with non-bursting entities people are likely to make similar mentions of an entity, discussing the events/organisations the entity is involved in. Figure 4.15 shows tweets about 'Roger Taylor which use similar mention texts and are made in relation to either his birthday or being in the band Queen. Storing fewer mentions about each entity frees up space to represent more entities in the sample.

```
Queen's Roger Taylor re-releases anti-Murdoch song - video:  Dear Mr.  Murdoch '
to be available from iTunes later
Happy Birthday , Sir Roger Taylor ...  Best drummer ever ...  Muahaha ...
So many birthdays today!  Wishing Roger Taylor of @QueenWillRock the Happiest of
Birthdays!!  All the best from us at Talenthouse :)
Great interview Brian May & Roger Taylor #Queen recorded last week @BBCRadio2 RT
if Queen fan!  Feel free to embed too!
```

Figure 4.15: Tweets about Roger Taylor, the drumber from Queen, mention text is coloured red.

The diversity technique improves upon the cache sampling techniques by first determining if a mention should update an existing mention, including mentions currently in the cache or replace a mention in the sample using a cache sampling technique as shown in Figure 4.16 (psudocode can be found in Appendix A.6). It would be possible to use a complex model to update an existing mention combining components from the mentions though to reduce any computational overhead and ensure recency is still modelled the current mention replaces the existing mention. Diversity sampling introduces another parameter: the similarity threshold for replacement.

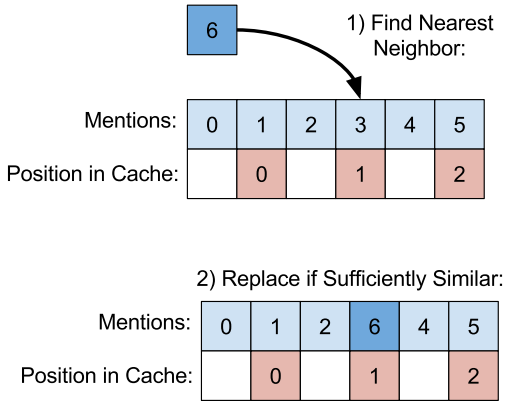

Figure 4.16: Demonstration of the replacement sampling techniques.

Figure 4.17 shows the running example with both Diversity-Cache sampling techniques, the example assumes the system perfectly resolves all the mentions. The Diversity-Cache-Win sampling guarantees mention 13 will be correctly resolved as mentions 9-12 replace mention 8 in the cache, freeing up space for mention 5 to remain in the sample. The techniques do not guarantee that mention 23 will be correctly resolved as mention 2 is pushed out of the cache when mention 13 is processed. As only two mentions are added to the sample (the two singletons) before mention 23 is processed. Mention 23 may be correctly resolved when using exponential sampling as when a mention is pushed out the cache it is not removed from the sample, there is a chance mention 2 remains in the sample to help resolve mention 23.

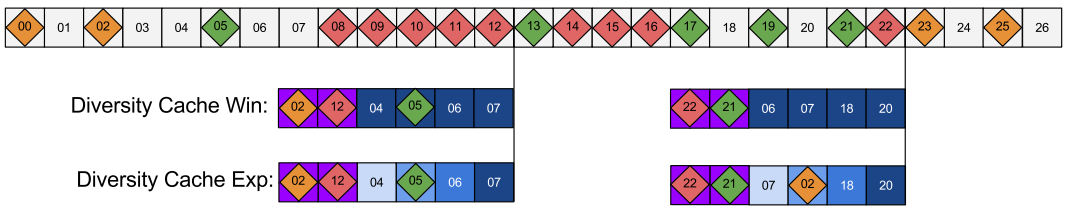

Figure 4.17: Running example demonstrating both diversity sampling techniques. Probability a mention is in the sample is indicated by background shade, lighter is lower probability. Purple mentions are in the cache.

Each sampling technique addresses some of the streaming CDC challenges outlined in section 4.3. Table 4.1 summarises the challenges addressed by each of the sampling techniques discussed. The diversity cache techniques address all the streaming CDC challenges. The new techniques presented have a strong focus on efficiency,

using a simple model to determine if a mention is likely to be used in the future and a replacement technique to update existing mentions in the sample. It is possible for more advanced versions of the techniques presented to perform better though at the expense of efficiency.

	Streaming CDC Challanges			
Sampling Technique	Recency	Distant Reference	Bursting Entities	Entity Diversity
Window	✓			
Biased	✓	✓	✓	
Uniform		✓	✓	
Cache-Win	✓	✓		
Cache-Exp	✓	✓		
Diversity-Cache-Win	✓	✓	✓	✓
Diversity-Cache-Exp	✓	✓	✓	✓

Table 4.1: The streaming CDC challanges addessed by each of the sampling techniques.

4.7 Experimental Framework

Experiments were performed using the TweetCDC corpus described in section 3.1. As some of the sampling techniques have parameters that need to be set, a streaming cross-fold evaluation technique was used (Levenberg and Osborne, 2009). The stream is split up into consecutive slices, and parameters are optimised (using grid search) on one slice and then evaluated on the next slice as shown in Figure 4.18. This evaluation setup maintains the streaming nature of the task where only the information prior to the start of the evaluation can be annotated.

The TweetCDC dataset is split into 11 constant-sized slices, each representing approximately one week. The first slice was used for development and setting the linking threshold; the remaining 10 slices are used for evaluation. For all techniques with a randomised component an average over 11 runs is reported.

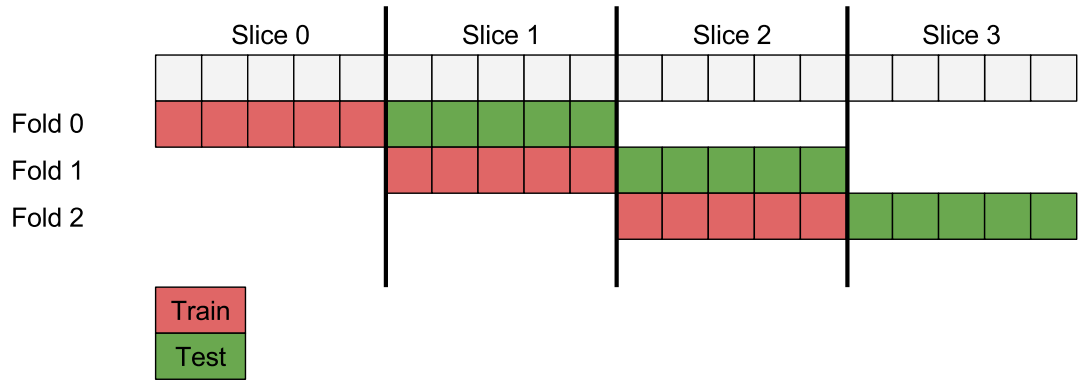

Figure 4.18: The cross fold evaluation setup used.

As the sample size will have a large impact on system performance, experiments were performed using various sample sizes. The sample sizes are all related to the average amount of mentions per day (79,761). To determine if the observed performance improvement is significant, a Wilcoxon signed rank test is used with a p-value of 5%.

4.8 Results

The performance results are summarised in Table 4.2. None of the existing sampling techniques significantly out-perform the simplest (window) sampling technique, with uniform sampling consistently performing worse.

	SS = 19K (0.25 Days)			SS=39K (0.50 Days)		
Technique	P	R	F	P	R	F
Window	25.6	69.1	37.3	32.7	70.8	44.7
Biased	25.5	69.1	37.3	33.1	71.1	45.2
Uniform	23.8	68.0	35.2	31.3	70.3	43.3
Cache-Win	28.1	69.3	40.0 $*$	34.6	70.8	46.5 $*$
Cache-Exp	28.2	69.1	40.1 $*$	35.9	71.3	47.7 $*$
D-C-Win	**39.2**	**73.8**	**51.2** $*$ †	**46.7**	**75.4**	**57.7** $*$ †
D-C-Exp	38.8	73.5	50.8 $*$ †	45.8	74.8	56.8 $*$ †
	SS = 59K (0.75 Days)			SS=79K (1.00 Days)		
Technique	P	R	F	P	R	F
Window	38.0	71.9	49.8	41.9	73.0	53.2
Biased	38.3	72.4	50.1	42.5	73.3	53.8
Uniform	36.3	71.3	48.1	40.2	72.5	51.7
Cache-Win	41.4	72.7	52.8 $*$	43.9	73.0	54.9 $*$
Cache-Exp	41.3	72.5	52.6 $*$	44.9	73.3	55.7 $*$
D-C-Win	**51.7**	**76.3**	**61.6** $*$ †	**53.8**	**76.3**	**63.1** $*$ †
D-C-Exp	50.8	75.8	60.3 $*$ †	52.8	75.9	62.3 $*$ †
	SS = 119K (1.50 Days)			SS=159K (2.00 Days)		
Technique	P	R	F	P	R	F
Window	47.7	74.1	58.0	52.0	74.9	61.4
Biased	48.2	74.3	58.5	52.1	74.8	61.4
Uniform	46.8	73.7	57.3	50.8	74.4	60.4
Cache-Win	49.6	74.2	59.5 $*$	53.0	75.0	62.1 $*$
Cache-Exp	50.0	74.0	59.7 $*$	52.8	74.5	61.8 $*$
D-C-Win	**57.1**	**76.7**	**65.5** $*$ †	**59.0**	**76.7**	**66.6** $*$ †
D-C-Exp	56.4	76.6	65.0 $*$ †	58.2	76.4	66.1 $*$ †
	SS≈600K (7.00 Days)					
Technique	P	R	F			
Sequental (No Sampling)	61.8	75.7	68.0			

Table 4.2: CEAFe performance for various sample sizes (SS) and sampling techniques. $*$ indicates significant improvement over Window sampling. † indicates significant improvement when the diversity technique is included.

None of the existing techniques significantly out-perform window sampling (though biased sampling performed slightly better) whereas the new techniques all significantly out-perform window sampling. The cache sampling technique performs significantly better than window sampling. Adding the diversity technique to the cache sampling results in another significant performance improvement.

Figure 4.19 shows how Window and Diversity Cache Window Sampling compare for the various sample sizes (amount of memory used) demonstrating the clear and consistent performance improvement. It is also clear that performance consistently improves as the sample size gets larger (more memory is used) for all systems.

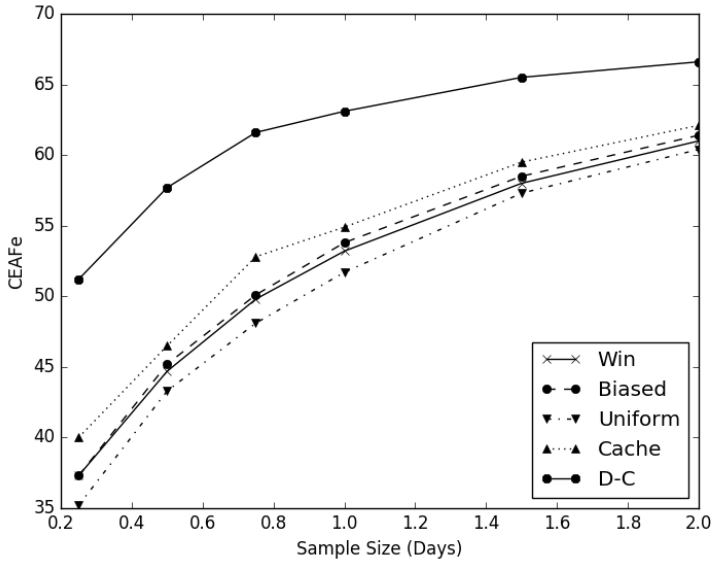

Figure 4.19: Performance of the sampling techniques for various sample sizes. Cache and Diversity-Cache (D-C) both use the Win variant as it performed better than Exp

4.9 Analysis

The sampling techniques are analysed with respect to the streaming CDC challenges outlined in section 4.3. All the following experiments were performed using a sample size of one day (79,761 mentions), optimising parameters on the first slice and then testing on the remainder of the dataset. Cache-Exp and Diversity-Cache-Exp are not reported as Cache-Win and Diversity-Cache-Win performed better. All results are averages over 11 runs.

4.9.1 Reference Age

Resolving distant references is particularly challenging in the streaming model of computation due to the need to forget some information. If a distant reference is resolved, an 'old' mention is identified as a nearest neighbour. To determine if distant references are being resolved the age of the mention identified as the nearest neighbour (reference age) is recorded. The average and maximum reference age for the sampling techniques are reported in table 4.3.

Technique	Average (Days)	Maximum (Days)
Sequential	13.8	76.9
Window	0.4	1.7
Biased	1.0	16.2
Uniform	16.8	77.0
Cache-Win	0.9	4.0
Diversity-Cache-Win	0.4	17

Table 4.3: The average and maximum reference age for the various the sampling techniques.

The average and maximum resolution age of Biased is larger than Window, clearly demonstrating that some of the older mentions are being used to resolve distant references.

Uniform has a large average and maximum resolution age, very similar to the sequential system. Distant references are clearly being resolved.

Compared to Window sampling, Cache-Win increases the average and maximum resolution ages, demonstrating that older mentions are being kept in the sample and they are being used to resolve mentions.

With Diversity-Cache-Win, the average is the same as Window sampling although the maximum is much larger. The mentions being used are the ones that have replaced existing mentions reducing the average reference age. This frees up space for distant references to be resolved when required as indicated by the high maximum reference age.

4.9.2 Average Age of Mentions in Sample

The average age of the mentions in the sample provides another measure of how well the sampling technique models recency/distant reference. Plotting the average age as the system processes the stream highlights how events such as bursting entities affect the system's ability to resolve distant references as shown in Figure 4.20. The wavy pattern is due to the 24 hour period of the underlying people's use of Twitter.

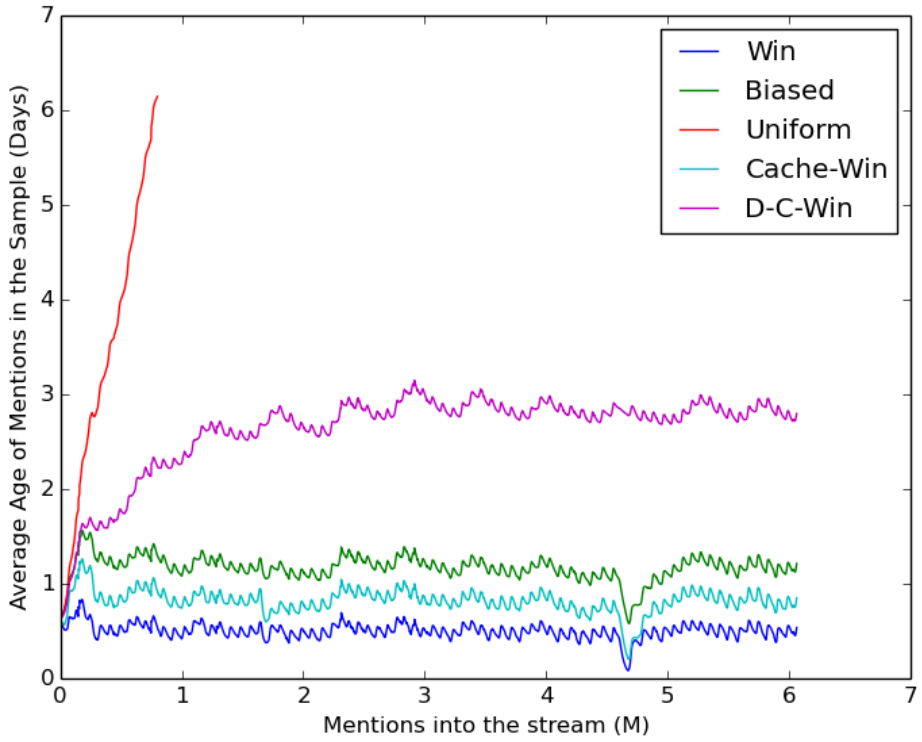

Figure 4.20: Average age of mentions in the sample for various sampling techniques. The line for Uniform sampling was truncated for clarity.

Uniform sampling stores the oldest mentions, increasing linearly as expected for a uniform sample (its line was truncated for clarity). Biased achieves a modest increase in the average age of the sample compared to window.

The average age of the Cache-Win sample is consistently between biased and window demonstrating that older mentions are being stored.

The Diversity-Cache-Win sampling technique stores the oldest mentions by a large margin, and the space saved by replacing similar mentions allows mentions to stay in the sample for much longer.

At approximately 4.6M mentions into the stream, the average age of all samples apart from Diversity-Cache-Win drop. This is due to a bursting entity demonstrating that the diversity sampling technique limits the impact of bursting entities.

4.9.3 Amount of Entities Represented in Sample

The sample should contain a diverse range of mentions representing a large amount of entities. Computing the amount of entities represented in the sample directly measures this. As the TweetCDC dataset contains only partial annotations, the sequential system is used to predict the entity being referred to for all the mentions, allowing an estimate of the amount of entities represented in the sample to be plotted in Figure 4.21.

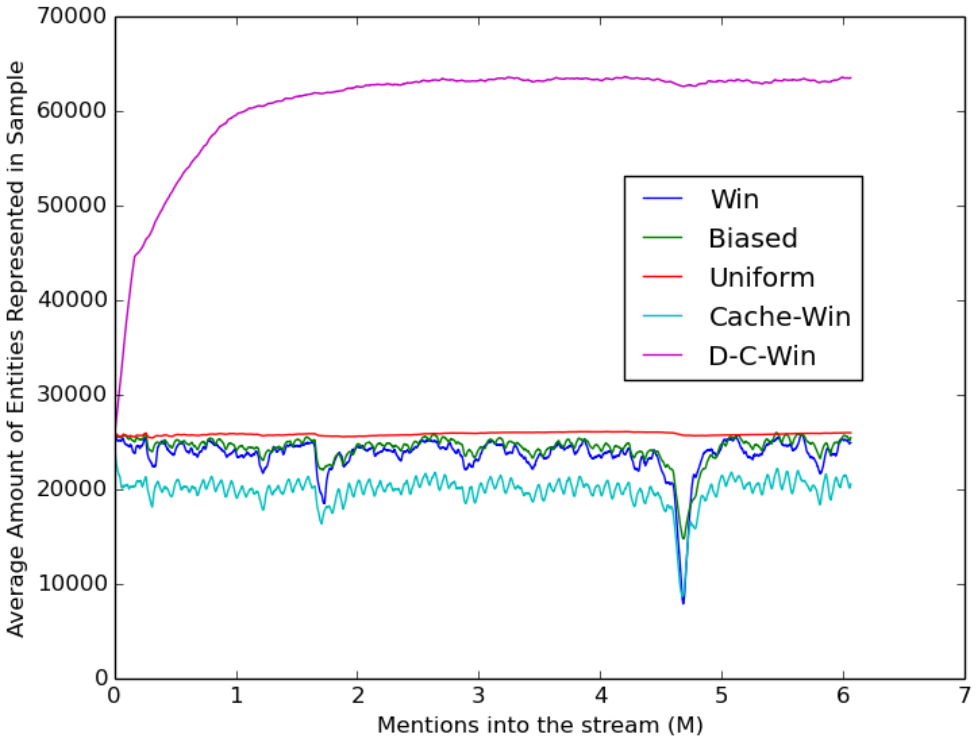

Figure 4.21: The amount of entities represented in the sample for the various sampling techniques.

Window and Biased sampling follow a similar shape maintaining approximately 25K entities. There are two clear dips in the diversity at approximately 1.7M and 4.6M mentions into the stream, this is caused by two bursting entities. The first bursting entity was not clearly evident in Figure 4.20 as it was a much smaller burst.

Uniform Sampling maintains a fairly constant amount of entities. Sampling a temporally diverse set of mentions means that temporal effects such as bursting entities should be minimal.

Cache-Win sampling stores fewer entities than the other techniques. Multiple mentions of the same entity are identified as likely to be useful again in the future. It is only when the diversity technique is introduced that is there a clear improvement in the amount of entities represented in the sample.

4.10 Summary

In this chapter, the first truly streaming CDC system was introduced, it conforms to the streaming model of computation introduced in Section 2.4.1 by storing a constant sized sample of previously encountered mentions.

The challenges faced by a streaming CDC were identified in section 4.3: Modelling recency/distant reference while ensuring bursting entities do not dominate and a broad range of entities are represented in the sample.

Existing sampling techniques were analysed with respect to the streaming CDC challenges, demonstrating they do not address all of them. New techniques were proposed and analysed with respect to the challenges, demonstrating a significant performance improvement when evaluated on the TweetCDC dataset from section 3.1.

Improving the sampling technique results in a significant performance improvement without the need for more memory. The new sampling techniques presented in this chapter achieve a performance within 95% of a non-streaming system while using 80% less memory.

Chapter 5

Conclusion

This thesis addressed a central problem for any entity level NLP, how to identify every entity being mentioned in a collection of documents. Failure to resolve each mention before any entity level NLP leads significantly reduces its effectiveness. For example: any information about 'Chris Evans' the actor will be mixed in with 'Chris Evans' the radio presenter. Information about the pop-star 'Snoop Dog' will be separate to information about 'Snoop Lion' despite them being the same person.

People want entity specific information in real time, so they are up to date with the broad range of entities that interest them. This means any system built for resolving mentions should guarantee instant resolution. This thesis addressed two important tasks for real-time CDC:

1. **Dataset Creation and Evaluation:** Existing techniques for dataset creation either introduce a significant amount of bias or require so much manual labour they are not applicable for large datasets. A new technique for creating a large CDC dataset with minimal compromises by annotating a sample of entities was introduced and used to create a new dataset. Existing evaluation measures are not applicable to datasets with only partial annotations hence a modification of an existing measure is introduced and experimentally verified.

2. **A Streaming Model for CDC:** A real-time CDC system must conform to the streaming model of computation: It must only use a constant amount of memory. A streaming CDC system was introduced that only stores a constant sized sample of previously seen mentions. Existing techniques to maintain a sample of mentions fail to address the challenges of streaming CDC. The new techniques presented in this thesis result in a significant performance improvement.

5.1 Future Work

The system presented in this thesis can be improved upon with a better mention similarity measure. With the current mention similarity measure, term similarity is only modelled for the target mention text. The other, associated mention texts, are compared using whole terms. By implementing a contextual similarity measure that includes a model of term similarity (such as soft-tf-idf) mention text similarity of associated mentions would also be modeled. Techniques that model term similarity are typically computationally expensive, building a scalable system for modelling term similarity in the streaming model of computation is a challenging problem.

Short social media posts often lack sufficient contextual information. It is trivial to find a tweet about an entity with no other mentions to provide context. This is particularly true for mentions of entities when there is some presumed context coming from some underlying event happening (such as sports events). Novel context sources must be used to help resolve these mentions. One interesting source of contextual information is temporal information. The time between successive mentions decays as shown in Chapter 4 although for some entities there are periodic patterns caused by daily or weekly events such as sports events and the news cycle. Modelling these patterns and using them to help resolve CDC should help improve performance. Modelling them within the confines of the streaming model of computation would be particularly challenging.

The streaming CDC system presented clearly has room for modifications. The cache component of the new sampling techniques uses a very simple model for determining if a mention is likely to be useful in the future. The diversity component replaces mentions instead up updating the representation. Simple improvements to these components has the potential for a significant performance improvement at the expense of computational efficiency.

Future work could also address some of the problems deploying CDC. How do humans want to interact with entity-specific information? It it possible to predict which information will be important? Is the best model for what information a user finds important user-specific or entity-specific?

Appendix A

Sampling Techniques Pseudocode

The following definitions are useful for understand the algorithms described in this appendix:

- S: Sample.
- k: Sample size.
- M: Stream of mentions.
- i: Amount of mentions processed.
- C: LIFO cache.
- l: Size of LIFO cache.
- θ_l: The linking threshold used to determine if two mentions refer to the same entity. Set before training.
- θ_r: The threshold for replacement used by the diversity technique.
- $Update\_Cache(C, l, m)$: If m is not in cache C add it removing a mention using the last in first out rule if the cache exceeds size l.
- $Update\_Win(S, C, m)$: Remove the oldest mention in sample S and not in cache C, add mention m.
- $Update\_Exp(S, C, m)$: Remove a mention uniformly at random from S that is not in cache C, add mention m.
- $Update\_Sample(S, C, m)$: Call $Update\_Win$ or $Update\_Exp$ depending on which technique is being used to maintain the sample.

A.1 Sequential (Non Sampling)

S: Sample

k: Sample Size

M: Stream of Mentions

i: Amount of Mentions Processed

```
foreach m in M do
|  S[i] = m
end
```

Algorithm 1: Pseudocode for Sequential Technique.

A.2 Window Sampling

S: Sample

k: Sample Size

M: Stream of Mentions

i: Amount of Mentions Processed

```
foreach m in M do
|  if i < k then
|  |  S[i] = m
|  else
|  |  S[i%k] = m
|  end
end
```

Algorithm 2: Pseudocode for Window Sampling Technique.

A.3 Biased Reservoir Sampling

S: Sample

k: Sample Size

M: Stream of Mentions

i: Amount of Mentions Processed

Paramiter: P_i Probability of insertion

foreach *m in M* **do**
    if $i < k$ **then**
        S[i] = m
    else
        $r \leftarrow Rand\_float(0,1)$
        if $r < P_i$ **then**
            $o \leftarrow Rand\_int(0,k)$
            S[o] = m
        end
    end
end

Algorithm 3: Pseudocode for Biased Reservoir Sampling Technique.

A.4 Uniform Reservoir Sampling

S: Sample

k: Sample Size

M: Stream of Mentions

i: Amount of Mentions Processed

foreach *m in M* **do**
    if $i < k$ **then**
        S[i] = m
    else
        $r \leftarrow Rand\_int(0,i)$
        if $r < k$ **then**
            S[r] = m
        end
    end
end

Algorithm 4: Pseudocode for Uniform Sampling Technique.

A.5 Cache Sampling

S: Sample

k: Sample Size

M: Stream of Mentions

i: Amount of Mentions Processed

C: LIFO Cache

Paramiter: l Size of Cache C.

Paramiter: θ_l Linking Threshold.

foreach *m in M* **do**
  if $i < k$ **then**
    S[i] = m
  else
    $m' \leftarrow Nearest\_Neigbhor(S,m)$
    $d \leftarrow sim(m,m')$
    if $d > \theta_l$ **then**
      $Update\_Cache(C,m')$
    end
    $Update\_Sample(S,C,m)$
  end
end

Algorithm 5: Pseudocode for Cache Sampling Technique.

A.6 Diversity Cache Sampling

S: Sample

k: Sample Size

M: Stream of Mentions

i: Amount of Mentions Processed

C: LIFO Cache

Paramiter: l Size of Cache C.

Paramiter: θ_l Linking Threshold.

Paramiter: θ_r Threshold for Replacement.

foreach *m in M* **do**

  if $i < k$ **then**

    S[i] = m

  end

  $m' \leftarrow Nearest\_Neigbhor(S,m)$

  $d \leftarrow sim(m,m')$

  if $d > \theta_l$ **then**

    $Update\_Cache(C,m')$

  end

  if $d > \theta_r$ **then**

    $m' \leftarrow m$

  else

    $Update\_Sample(S,C,m)$

end

Algorithm 6: Pseudocode for Diversity-Cache Sampling Technique.

Bibliography

Aggarwal, C. C. (2006). On biased reservoir sampling in the presence of stream evolution. In *Proceedings of the 32nd international conference on Very large data bases*, pages 607–618. VLDB Endowment.

Aggarwal, C. C., Han, J., Wang, J., and Yu, P. S. (2003). A framework for clustering evolving data streams. In *Proceedings of the 29th international conference on Very large data bases-Volume 29*, pages 81–92. VLDB Endowment.

Andrews, N., Eisner, J., and Dredze, M. (2014). Robust entity clustering via phylogenetic inference. In *Association for Computational Linguistics (ACL)*.

Bagga, A. and Baldwin, B. (1998a). Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–566. Citeseer.

Bagga, A. and Baldwin, B. (1998b). Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 79–85. Association for Computational Linguistics.

Baron, A. and Freedman, M. (2008). Who is who and what is what: Experiments in cross-document co-reference. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 274–283. Association for Computational Linguistics.

Bengtson, E. and Roth, D. (2008). Understanding the value of features for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 294–303. Association for Computational Linguistics.

Cai, J. and Strube, M. (2010). Evaluation metrics for end-to-end coreference resolution

systems. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 28–36. Association for Computational Linguistics.

Charikar, M., Chekuri, C., Feder, T., and Motwani, R. (1997). Incremental clustering and dynamic information retrieval. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 626–635. ACM.

Chen, Y. and Martin, J. (2007). Towards robust unsupervised personal name disambiguation. In *EMNLP-CoNLL*, pages 190–198. Citeseer.

Chen, Y. and Tu, L. (2007). Density-based clustering for real-time stream data. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM.

Cohen, W., Ravikumar, P., and Fienberg, S. (2003). A comparison of string metrics for matching names and records. In *KDD workshop on data cleaning and object consolidation*, volume 3, pages 73–78.

Cohen, W. and Richman, J. (2001). Learning to match and cluster entity names. In *ACM SIGIR-2001 Workshop on Mathematical/Formal Methods in Information Retrieval*.

Counter, T. (2016). Twitter Top 100 Most Followers. `http://twittercounter.com/pages/100`. Accessed: 2016-08-14.

Cucerzan, S. (2007). Large-scale named entity disambiguation based on wikipedia data. In *EMNLP-CoNLL*, volume 7, pages 708–716.

Durrett, G. and Klein, D. (2013). Easy victories and uphill battles in coreference resolution. In *EMNLP*, pages 1971–1982.

Dutta, S. and Weikum, G. (2015a). C3el: A joint model for cross-document co-reference resolution and entity linking. In *Conference on Empirical Methods in Natural Language Processing*, pages 846–856. ACL.

Dutta, S. and Weikum, G. (2015b). Cross-document co-reference resolution using sample-based clustering with knowledge enrichment. *Transactions of the Association for Computational Linguistics*, 3:15–28.

Edwards, J. (2016). Leaked Twitter API data shows the number of tweets is in serious decline. `http://uk.businessinsider.com/`

tweets-on-twitter-is-in-serious-decline-2016-2. Accessed: 2016-08-26.

Gooi, C. H. and Allan, J. (2004). Cross-document coreference on a large scale corpus. Technical report, DTIC Document.

Green, S., Andrews, N., Gormley, M. R., Dredze, M., and Manning, C. D. (2012). Entity clustering across languages. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL HLT '12, pages 60–69, Stroudsburg, PA, USA. Association for Computational Linguistics.

Guild, S. A. (2016). Getting Started as an Actor FAQ. https://www.sagaftra.org/content/getting-started-actor-faq. Accessed: 2016-07-21.

Guo, J., Xu, G., Cheng, X., and Li, H. (2009). Named entity recognition in auery. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 267–274. ACM.

Haghighi, A. and Klein, D. (2007). Unsupervised coreference resolution in a nonparametric bayesian model. In *Annual meeting-Asociation for Computational Linguistics*, page 848.

Holmes, S. (2016). Meet 3 Megafans Keeping Up With the Kardashians on Social Media. http://www.elle.com/culture/celebrities/news/a34965/kardashian-fan-accounts/. Accessed: 2016-08-14.

Lee, H., Recasens, M., Chang, A., Surdeanu, M., and Jurafsky, D. (2012). Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500. Association for Computational Linguistics.

Levenberg, A. and Osborne, M. (2009). Stream-based randomised language models for smt. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 756–764. Association for Computational Linguistics.

Lui, M. and Baldwin, T. (2012). langid. py: An off-he-shelf language identification tool. In *Proceedings of the ACL 2012 system demonstrations*, pages 25–30. Association for Computational Linguistics.

Luo, X. (2005). On coreference resolution performance metrics. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.

Mann, G. S. and Yarowsky, D. (2003). Unsupervised personal name disambiguation. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 33–40. Association for Computational Linguistics.

McGrath, F. (2015). A Fifth use Social Media to Follow Celebrities. `http://www.globalwebindex.net/blog/a-fifth-use-social-media-to-follow-celebrities`. Accessed: 2016-08-14.

Meyer, R. (2016). How Many Stories Do Newspapers Publish Per Day? `http://www.theatlantic.com/technology/archive/2016/05/how-many-stories-do-newspapers-publish-per-day/483845/`. Accessed: 2016-08-26.

Moreau, E., Yvon, F., and Cappé, O. (2008). Robust similarity measures for named entities matching. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 593–600. Association for Computational Linguistics.

Muthukrishnan, S. (2005). *Data streams: Algorithms and applications*. Now Publishers Inc.

Ng, V. and Cardie, C. (2002). Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 104–111. Association for Computational Linguistics.

Nguyen, D. B., Hoffart, J., Theobald, M., and Weikum, G. (2014). Aida-light: High-throughput named-entity disambiguation. In *LDOW*. Citeseer.

Niu, C., Li, W., and Srihari, R. K. (2004). Weakly supervised learning for cross-document person name disambiguation supported by information extraction. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 597. Association for Computational Linguistics.

O'callaghan, L., Mishra, N., Meyerson, A., Guha, S., and Motwani, R. (2002). Streaming-data algorithms for high-quality clustering. In *icde*, volume 2, page 685.

Osborne, M., Lall, A., and Van Durme, B. (2014). Exponential reservoir sampling for streaming language models. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 687–692. Association for Computational Linguistics.

Pedersen, T., Purandare, A., and Kulkarni, A. (2005). Name discrimination by clustering similar contexts. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 226–237. Springer.

Petrović, S., Osborne, M., and Lavrenko, V. (2010). Streaming first story detection with application to twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–189. Association for Computational Linguistics.

Philips, L. (2000). The double metaphone search algorithm. *C/C++ users journal*, 18(6):38–43.

Pradhan, S., Luo, X., Recasens, M., Hovy, E., Ng, V., and Strube, M. (2014). Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 30–35. Association for Computational Linguistics.

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., and Zhang, Y. (2012). Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 1–40. Association for Computational Linguistics.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. (2011). Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–27. Association for Computational Linguistics.

Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., and Manning, C. (2010). A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501. Association for Computational Linguistics.

Rahman, A. and Ng, V. (2009). Supervised models for coreference resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 968–977. Association for Computational Linguistics.

Rao, D., McNamee, P., and Dredze, M. (2010). Streaming cross document entity coreference resolution. In *Coling 2010: Posters*, pages 1050–1058. Coling 2010 Organizing Committee.

Ristad, E. S. and Yianilos, P. N. (1998). Learning string edit distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(5):522–532.

Ritter, A., Clark, S., Mausam, and Etzioni, O. (2011). Named entity recognition in tweets: An experimental study. In *EMNLP*.

Ritter, A., Etzioni, O., Clark, S., et al. (2012). Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1104–1112. ACM.

Singh, S., Subramanya, A., Pereira, F., and McCallum, A. (2011). Large-scale cross-document coreference using distributed inference and hierarchical models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 793–803. Association for Computational Linguistics.

Singh, S., Subramanya, A., Pereira, F., and McCallum, A. (2012). Wikilinks: A large-scale cross-document coreference corpus labeled via links to wikipedia. *University of Massachusetts, Amherst, Tech. Rep. UM-CS-2012-015*.

Soon, W. M., Ng, H. T., and Lim, D. C. Y. (2001). A machine learning approach to coreference resolution of noun phrases. *Computational linguistics*, 27(4):521–544.

Stoyanov, V., Gilbert, N., Cardie, C., and Riloff, E. (2009). Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 656–664. Association for Computational Linguistics.

Strassel, S., Przybocki, M. A., Peterson, K., Song, Z., and Maeda, K. (2008). Linguistic resources and evaluation techniques for evaluation of cross-document automatic content extraction. In *LREC*.

Tong, S. and Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66.

Vilain, M., Burger, J., Aberdeen, J., Connolly, D., and Hirschman, L. (1995). A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52. Association for Computational Linguistics.

Vitter, J. S. (1985). Random sampling with a reservoir. *ACM Transactions on Mathematical Software (TOMS)*, 11(1):37–57.

Von Luxburg, U., Williamson, R. C., and Guyon, I. (2012). Clustering: Science or art? In *ICML Unsupervised and Transfer Learning*, pages 65–80.

Weischedel, R., Palmer, M., Marcus, M., Hovy, E., Pradhan, S., Ramshaw, L., Xue, N., Taylor, A., Kaufman, J., Franchini, M., et al. (2011). Ontonotes release 4.0. *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium*.

Wick, M., Singh, S., and McCallum, A. (2012). A discriminative hierarchical model for fast coreference at large scale. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 379–388. Association for Computational Linguistics.

Zobel, J. and Dart, P. (1995). Finding approximate matches in large lexicons. *Software: Practice and Experience*, 25(3):331–345.